Volume 4, Issue 2, February-2017, pp. 43-47 ISSN (O): 2349-7084

International Journal of Computer Engineering In Research Trends

Available online at: www.ijcert.org

Music Genre Classification Using MFCC, K-NN and SVM Classifier

Nilesh M. Patil[1], Dr. Milind U. Nemade[2],

Ph.D Research Scholar[1],

Pacific Academy of Higher Education and Research University, Udaipur, India.

Email ID: nileshdeep@gmail.com.

Professor & Head[2],

Department of Electronics Engineering, K J Somaiya Institute of Engineering & Information Technology, Mumbai.

Email ID: munemade@gmail.com

Abstract:-The audio corpus available today on Internet and Digital Libraries is increasing rapidly in huge volume. We need to properly index them if we want to have access to these audio data. The search engines available in market also find it challenging to classify and retrieve the audio files relevant to the user's interest. In this paper, we describe an automated classification system model for music genres. We firstly found good feature for each music genre. To obtain feature vectors for the classifiers from the GTZAN genre dataset, features like MFCC vector, chroma frequencies, spectral roll-off, spectral centroid, zero-crossing rate were used. Different classifiers were trained and used to classify, each yielding varying degrees of accuracy in prediction.

Keywords: Music, MFCC, K-NN, SVM, GTZAN dataset.

---

1. Introduction:

Music classification is an interesting problem with varying applications from Drinkify to Pandora. Music classification is still considered to be one of the research area due to the challenge in selection and extraction of optimal audio features. Music genre classification has been a challenging task in the field of Music Information Retrieval (MIR). Music genres are inherently subjective due to which they are hard to systematically and consistently describe. Genre classification, till now, had been done manually by concatenating it to metadata repository of audio files. This paper however aims at content-based classification, focusing on information within the audio. We used traditional machine learning approach for classification by finding suitable features of audio signals, training classifier on feature data and make predictions..

2. Related Works

Tzanetakis and Cook [1] pioneered the work on music genre classification using machine learning technique. They created the GTZAN dataset and is to date considered as a standard for genre classification. Changsheng Xu et al. [2] have shown how to use support vector machines (SVM) for this task. Authors used supervised learning approaches for music genre classification. Scaringella et al. [3] gives a comprehensive survey of both features and classification techniques used in the music genre classification. Riedmiller [4] used unsupervised learning creating a dictionary of features.

3. Description

An open source software framework called MARSYAS (Music Analysis, Retrieval, and Synthesis for Audio Signals) is available for audio processing with specific emphasis on Music Information Retrieval Applications [6]. MARSYAS website gives access to GTZAN dataset which is a collection of 900 audio tracks each 30 seconds long. There are 9 genres represented, each containing 100 tracks. All the tracks are 22050Hz Mono

16-bit audio files in .au format. We first convert the audio from .au format to .wav format so as to make it compatible to Python's wave module for reading audio files. For this conversion we use the open source SoX [5] utility. To classify our audio clips, we choose 5 features namely MFCC, spectral centroid, zero crossing rate, Chroma frequencies, spectral roll-off. These 5 features are concatenated to give a 28 length feature vector. Then we use different multiclass classifiers to obtain our result.

4. Methodology

The first step we perform is feature extraction. The five features used to create a single feature vector are described in this section.

1. Mel-Frequency Cepstral Coefficients (MFCC): These are a set of short term power spectrum characteristics of audio files. It models the characteristics of human voice. We are taking into consideration 13 coefficients as the part of the final feature vector. The method to implement this feature vector is shown in figure 1.

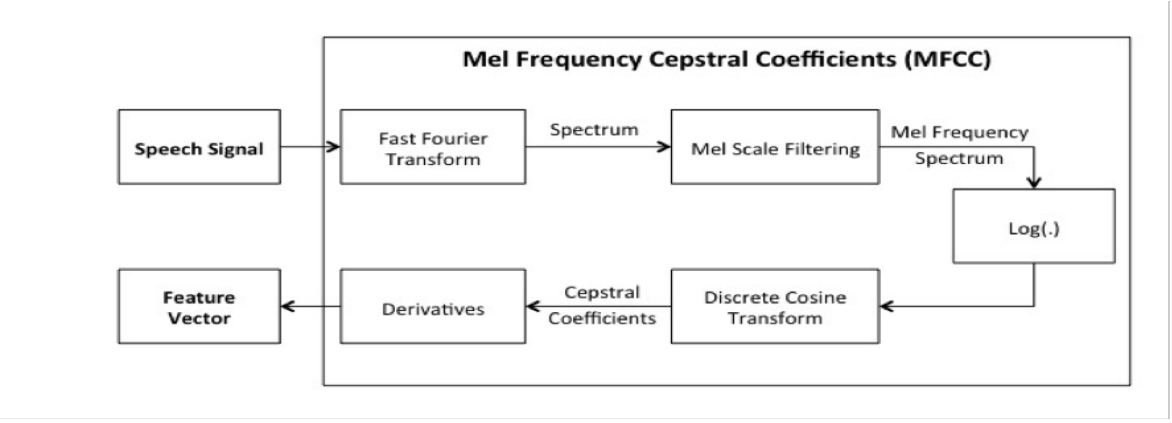

Figure 1 Steps in MFCC

- We divide the signal into several short frames so as to keep an audio signal constant.
- For each frame, we calculate periodogram estimate of the power spectrum to know frequencies present in the short frames.
- Push the power spectra into the mel filterbank and collect the energy in each filter to sum it. With this we get the energy existing in the various frequency regions. The formula to work with mel scale is:

$$M(f) = 1125 \ln\left(1 + \frac{f}{700}\right) \quad (1)$$

- We then calculate the logarithm of the filterbank energies. This enables humans to have features closer to what they can hear.
- Calculating the Discrete Cosine Transform (DCT) of the result decorrelates the filterbank energies with each other.
- We just keep first 13 DCT coefficients discarding the higher DCT coefficients that can introduce errors by representing changes in the filterbank energies.

2. Chroma Frequencies: Chroma frequency vector discretizes the spectrum into chromatic keys, and represents the presence of each key. We take the histogram of present notes on a 12-note scale as a 12 length feature vector. The chroma frequencies have a music theory interpretation. The histogram over the 12-note scale actually is sufficient to describe the chord played in that window. It provides a robust way to describe a similarity measure between music pieces.

3. Spectral Centroid: This parameter characterizes the spectrum of the signal. It indicates the location of the 'centre of gravity' of the magnitude spectrum. Perceptually, it gives the impression of 'brightness' of a sound. It can be evaluated as the weighted mean of the spectral frequencies. We find the FFT of the signal segment and find the average energy weighted by sum of spectrum amplitudes within one frame. The spectral centroid measures the spectral energy distribution in easy-state portions of tone. It measures the spectral shape. Higher centroid values correspond to 'brighter' textures with more high frequencies.

$$Spectral\ Centroid = \frac{\sum_{n=0}^{N-1} f(n)x(n)}{\sum_{n=0}^{N-1} x(n)} \quad (2)$$

Here, *f(n)* is the magnitude of the FFT for a frame *n* and *x(n)* is the index of the frequency bin

4. Spectral Roll-off: It is defined as the frequency bin M below which 85% of the magnitude distribution is concentrated. This is one more measure of spectral shape.

$$\sum_{n=0}^{M} f(n) = 0.85 * \sum_{n=0}^{N} f(n) \qquad (3)$$

It represents the frequency at which high frequencies decline to 0.

5. Zero Crossing Rate: It represents the number of times the waveform crosses 0. It usually has higher values for highly percussive sounds like those in metal and rock.

The second step we perform is the classification. Once the feature vectors are obtained, we train different classifiers on the training set of feature vectors. Following are the different classifiers that were used: K Nearest Neighbors, Linear Kernel SVM, and Polynomial Kernel SVM. The KNN algorithm is among the simplest of all machine learning algorithms. KNN classifier is a type of instance based learning technique and predicts the class of a new test data based on the closest training examples in the feature space. KNN is a variable-bandwidth, kernel density estimator with a uniform kernel. SVM is a very useful technique used for classification. It is a classifier which performs classification methods by constructing hyper planes in a multidimensional space that separates different class labels based on statistical learning theory. We have used linear kernel and polynomial kernel SVM classifiers.

The support vector classifier is a natural approach for classification in the two-class setting if the boundary between them is linear. Similarly, when we extend the linear model to account for non-linear relationships, we can transform the predictors using quadratic, cubic, and even higher order polynomial functions. The parameters used for various classifiers were obtained by manual tuning. It was observed that any single classifier did not classify all the genres well. For example in the SVM with polynomial kernel worked well for most genres except blues and rock (See figure 2). This could have been due to the fact that many other genres are derived from blues.

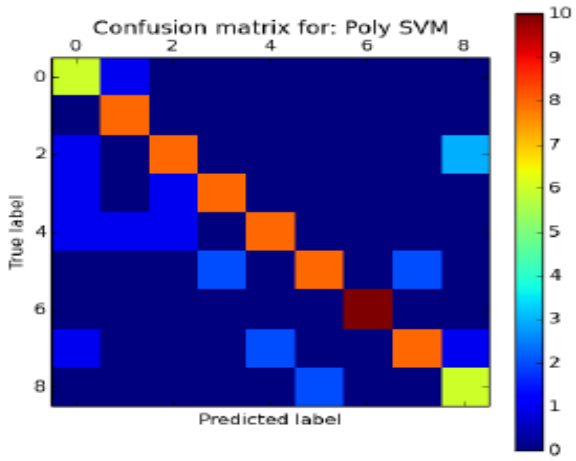

Figure 2 Confusion Matrix for Polynomial Kernel SVM Classifier

5. Results and Discussion

We measure the performance of the system using metrics like accuracy, recall and precision. Accuracy is defined as the ratio of number of correctly classified results to the total number of the classified results. Precision is defined as the ratio of the number of correct results to the number of predicted results. Recall is defined as the ratio of the number of correct results to the number of returned results. Higher the value of recall and precision will give better efficiency in classification. The genre-wise precision, recall and f-1 scores for different classifiers are given below.

Genre	Nearest Neighbors				Linear SVM				Poly SVM			
	Precision	Recall	F1-Score	Support	Precision	Recall	F1-Score	Support	Precision	Recall	F1-Score	Support
Blues	0.70	0.64	0.67	11	0.10	0.33	0.15	3	0.60	0.86	0.71	7
Classical	0.70	0.78	0.74	9	0.90	0.90	0.90	10	0.80	1.00	0.89	8
Country	0.90	0.53	0.67	17	0.80	0.57	0.67	14	0.80	0.67	0.73	12
Disco	0.50	0.56	0.53	9	0.50	0.71	0.59	7	0.80	0.80	0.80	10
Jazz	0.40	1.00	0.57	4	0.90	0.75	0.82	12	0.80	0.73	0.76	11
Metal	0.80	0.62	0.70	13	0.80	0.53	0.64	15	0.80	0.67	0.73	12
Pop	1.00	0.91	0.95	11	1.00	0.91	0.95	11	1.00	1.00	1.00	10
Reggae	0.40	0.50	0.44	8	0.30	0.33	0.32	9	0.80	0.67	0.73	12
Rock	0.40	0.50	0.44	8	0.10	0.11	0.11	9	0.60	0.75	0.67	8
Avg/Total	**0.70**	**0.64**	**0.66**	**90**	**0.68**	**0.60**	**0.63**	**90**	**0.79**	**0.78**	**0.78**	**90**

The predictions obtained after training each classifier are given below.

Nearest Neighbors	Linear SVM	Poly SVM
[[7 1 1 0 1 1 0 0 0] [0 7 0 0 2 0 0 0 0] [0 1 9 0 0 0 0 3 4] [2 0 0 5 0 0 0 1 1] [0 0 0 0 4 0 0 0 0] [0 0 0 2 0 8 0 2 1] [0 0 0 1 0 0 10 0 0] [1 1 0 0 2 0 0 4 0] [0 0 0 2 1 1 0 0 4]]	[[1 0 0 1 0 0 0 0 1] [1 9 0 0 0 0 0 0 0] [0 0 8 0 0 0 0 2 4] [0 0 0 5 0 0 0 0 2] [2 0 1 0 9 0 0 0 0] [0 1 1 2 0 8 0 2 1] [0 0 0 0 0 0 10 1 0] [4 0 0 0 1 0 0 3 1] [2 0 0 2 0 2 0 2 1]]	[[6 1 0 0 0 0 0 0 0] [0 8 0 0 0 0 0 0 0] [1 0 8 0 0 0 0 0 3] [1 0 1 8 0 0 0 0 0] [1 1 1 0 8 0 0 0 0] [0 0 0 2 0 8 0 2 0] [0 0 0 0 0 0 10 0 0] [1 0 0 0 2 0 0 8 1] [0 0 0 0 0 0 2 0 0 6]]
CORRECT PREDICTIONS: **58**	CORRECT PREDICTIONS: **54**	CORRECT PREDICTIONS: **70**
TOTAL PREDICTIONS: **90**	TOTAL PREDICTIONS: **90**	TOTAL PREDICTIONS: **90**
ACCURACY: **0.644444444444**	ACCURACY: **0.6**	ACCURACY: **0.777777777778**

Table 1 gives the statistics of these 3 classifiers and figure 3 gives pictorial representation of it.

Table 1 Statistics of classifier

Classifier	Mean Accuracy	Mean Precision	Mean Recall
K-NN	0.64	0.70	0.40
Linear Kernel SVM	0.60	0.68	0.60
Poly Kernel SVM	0.78	0.79	0.78

Classification accuracy varied between the different machine learning techniques and genres. However, polynomial kernel SVM proved to be more efficient giving accuracy of 78%, precision of 79% and recall of 78%.

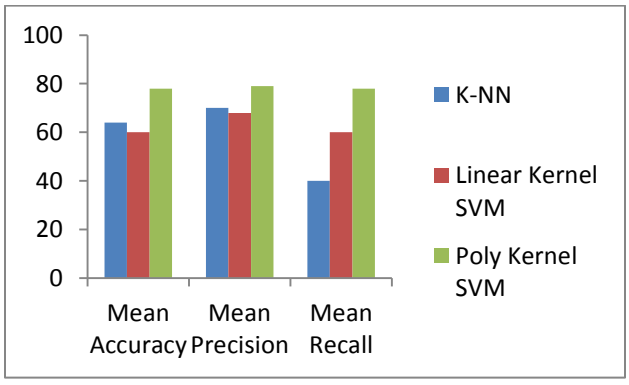

Figure 3 Pictorial representations of classifier statistics

6. Conclusion

We presented an automated system for music genre classification. MFCC features, Chroma features, spectral centroid, spectral roll-off, ZCR are used as the feature vectors and trained the system using three classifiers like k-NN, linear and polynomial kernel SVMs. We found polynomial kernel SVM to be better classifier giving accuracy of 78%.

7. References

[1] G. Tzanetakis, P. Cook, "Musical genre classification of audio signals", IEEE Transactions on Speech and Audio Processing, Vol. 10, Issue 5, July 2002.

[2] Chandsheng Xu, Mc Maddage, Xi Shao, Fang Cao, and Qi Tan, "Musical genre classification using support vector machines", IEEE Proceedings of International Conference of Acoustics, Speech, and Signal Processing, Vol. 5, pp. V-429-32, 2003.

[3] N. Scaringella, G. Zoia, and D. Mlynek, "Automatic genre classification of music content: a survey", IEEE Signal Processing Magazine, Vol. 23, Issue 2, pp. 133–141, 2006.

[4] Jan Wülfing and Martin Riedmiller, "Unsupervised learning of local features for music classification" ISMIR, pp. 139–144, 2012.

[5] Sox.sourceforge.net. Sox - sound exchange—homepage, 2015.

[6] http://marsyas.info/downloads/datasets.html

About the authors

Nilesh M. Patil is working as Assistant Professor in MCT's Rajiv Gandhi Institute of Technology, Mumbai. He earned B.E. & M.E. in Information Technology from Mumbai University, Maharashtra, India. He has 10.5 years of teaching experience. He is currently pursuing PhD in Computer Engineering from Pacific University (PAHER), Udaipur. His research interest includes Speech and Audio processing, Image Processing, Network Security, Data Mining, Big Data and IoT. He has published 10 research papers in International Journals and presented 8 research papers in National and International Conferences.

Dr. Milind U. Nemade was born in Maharashtra, India 1974. He graduated from the Amaravati University, Maharashtra, India in 1995. He received M.E (Electrical) degree with specialization in Microprocessor Applications in 1999 and Ph.D (Electrical) in 2010 from M.S. University of Baroda, Gujrat, India. Now he is Professor and Head of Electronics Engineering Department in K.J. Somaiya Institute of Engineering and Information Technology Sion, Mumbai, University of Mumbai, India. He presented and published four papers in national conferences, ten papers in international journals. His research interest includes Speech and Audio processing

