



Music Genre Classifier

29.04.2017

Gangesh Gudmalwar (140050058)

Tejesh Raut (140050008)

Himanshu Payal (140050011)

Project description

An extraordinary range of information is hidden inside of music waveforms, ranging from perceptual to auditory which inevitably makes large-scale applications challenging.

Our project focuses on large-scale data mining of music information with several datasets including Million Song Dataset Bertin-Mahieux et al., 2011), 1 which consists of 300GB of audio features and metadata.

We propose a cross-modal retrieval framework to combine the music and textual data for the task of genre classification

.

Approaches

We use MFCC-like features from Million Song Dataset .

Now using multiclass perceptron classifier:

A Perceptron keeps a weight vector $w(y)$ corresponding to each class y (next y is a suffix). Given a feature list f (a vector with the same number of dimensions as the weight vectors), the perceptron computes the class y whose weight vector is most similar to the input vector f . Formally, given a feature vector f , we score each class y with:

$$\text{score}(f, y) = \sum f(i)w(i)$$

In the basic multiclass Perceptron, we scan over the training data one instance at a time. When we come to an instance (f, y) we find the label with highest score:

$$y' = \arg\max_{y'} \{\text{score}(f, y')\}$$

breaking ties arbitrarily. We compare y' to the true label y . If $y' = y$, we have correctly classified the instance, and we do nothing. Otherwise, we predicted y' when we should have predicted y . That means that y has scored f lower, and/or $w_{y'}$ has scored f

higher, than what would have been ideal. To avert this error in the future, we update these two weight vectors accordingly:

$$w(y) = w(y) + n * f \text{ and}$$

$$w(y') = w(y') - n * f$$

where n is the learning rate.

Then we are going to use K-NN classification using Mahalanobis distance with KL-divergence.

Also, we will compare the results of K-NN with Multiclass SVM classification using 2-class SVM decision tree..

Methods/Algorithms

We created a neural network using standard pyrenn library from <https://pyrenn.readthedocs.io/en/latest/> but the results get hanged.

So we hard coded standard neural network.

Feature Design

There were 24 features present already. For training we removed the trackID. It consists of average timbre for 12 time intervals and variance of timbre for the same 12 time intervals. Output is one of the 10 genre classifiers. We changed the number of columns in output from 1 to 10. Every unique genre has a different column, with value set to 1 if that genre is present otherwise 0.

Team Contribution:

Tejesh(140050008): 100%

Gangesh(140050058): 100%

Himanshu(140050011): 100%

Data sets

<https://labrosa.ee.columbia.edu/millionsong/blog/11-2-28-deriving-genre-dataset>

- the following 10 ones that loosely mimic the GZTAN genres:

classic pop and rock, folk, dance and electronica, jazz and blues, soul and reggae,
punk, metal, classical, pop, hip-hop

Acknowledgements

<http://www.ee.columbia.edu/~dliang/files/FINAL.pdf>

<http://goo.gl/7weSmA>

<https://pyrenn.readthedocs.io/en/latest/>