MUSIC CLASSIFICATION USING HIGH-LEVEL MODELS

N. Wack, C. Laurier, O. Meyers, R. Marxer, D. Bogdanov, J. Serrà, E. Gomez, P. Herrera
Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain
{nicolas.wack,cyril.laurier,owen.meyers,ricard.marxer,
dmitry.bogdanov,joan.serraj,emilia.gomez,perfecto.herrera}@upf.edu

ABSTRACT

We report here about our submissions to different music classification tasks for the MIREX 2010 evaluations. These submissions are similar to the ones sent at MIREX 2009 (see [1]), if we look at the classifiers and the main audio features. However we added high-level features (or semantic features), based on Support Vector Machine models of curated databases of different kind. We submitted two different algorithms evaluated on Mood, Genre and Artists classification. One of them is a classification algorithm using a weighted sum of Support Vector Machines. The other one is based on distances (Euclidean in a reduced space using RCA and Kullback Leibler on Mel Frequency Cepstrum Coefficients), together with K-NN.

1. FEATURE EXTRACTION

This submission is coded in C++ and python. For the feature extraction part, we use an internal library of the Music Technology Group called Essentia [2]. This library contains all the features mentioned below. All frame-based statistics are aggregated using : mean and derivatives until second order, variance and derivatives until second order, minimum and maximum. We divide our features in two main categories. The "base" features which are state-of-the-art MIR features and the "high-level" features.

1.1 Base features

In Table 2 is the set of base features that performed the best in our preliminary experiment made on our genre, artist and mood databases.

1.2 High-level features

One of the originality of our approach is the integration of high-level (or semantic) descriptors. Low level features are convenient and easy to extract. They provide satisfying classification results in many tasks. However, high-level concepts encapsulate different pattern of low-level descriptors into a single representation that can add useful information. Based on this idea, we added high level features of different categories. These models are pre-trained algorithms using Support Vector Machines that are added to our bag of features. We consider them as other features with value between 0 and 1 corresponding to the SVM model prediction probability. Here we list the different models used:

Type	Features
Low level	barkbands spread, skewness, kurtosis, dissonance, hfc pitch and confidence, pitch salience, spectral complexity spectral crest, spectral decrease, energy, spectral flux spec spread/skewness/kurtosis, spec rolloff, strong peak ZCR, barkbands, mfcc, spectral contrast
Rhythm	bpm, beats loudness, onset rate
Sound FX	inharmonicity, odd2even, pitch centroid, tristimulus
Tonal	chords strength (frame), key strength(global), tuning freq

Table 1. Feature set for all our classifiers.

Type	Classes
Genre (1)	blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, rock
Genre (2)	alternative, electronic, funk/soul/rnb, pop, rock, blues, folk/country, jazz, rap/hiphop
Genre (3)	ambient, drum and bass, house, techno, trance
Genre (4)	classical, dance, hiphop, jazz, pop, rhythm and blues, rock, speech
Genre (5)	cha cha cha, quickstep, rumba-international, rumba-american, rumba-misc, tango, waltz, samba, viennese waltz, jive
Perceptual Speed	fast, medium, slow
Timbre	bright, dark
Culture	
Live / Studio	live, studio
Gender	male, female
Mood (5 classes)	5 classes similar to the mirex clusters [3]
Mood (Happy)	happy, not happy
Mood (Sad)	sad, not sad
Mood (Relaxed)	relaxed, not relaxed
Mood (Aggressive)	aggressive, not aggressive
Acoustic	acoustic, not acoustic
Electronic	electronic, not electronic

Table 2. High-level features. Types and classes of the SVM models trained on reference databases

2. CLASSIFICATION

The three classification algorithms are coded in C++ and python. They are implemented using Gaia, a library for manipulating dataset and computing similarity distances [2].
GRID SEARCH ????

2.1 Support Vector Machines

Support Vector Machines [4], is a widely used supervised learning classification algorithm. In a previous MIREX classification task (Audio Mood Classification), we submitted an algorithm based on SVMs that performed relatively well [5]. Indeed, most of the best ranked algorithm for classification use a SVM as a classifier. For our algorithm, we tried different kernel methods: linear, polynomial, radial basis function (RBF) and sigmoid. In our preliminary evaluations, we found that the better and more robust kernel is the RBF. Even if a RBF kernel is not always recommended for large feature sets compared to the size of the dataset [4], we had a good accuracy using this kernel for all tasks. It may not be the best solution always, but offer a good compromise in average. This algorithm is based on an implementation of Support Vector Machines called libsvm [6].

2.2 Relevant Component Analysis and Nearest Neighbours

Relevant Component Analysis (RCA) is a supervised transformation which aims at maximizing the global variance of a dataset while reducing the intra-class variance (representing unwanted variability). The algorithm is split in two parts: the first part is the dimensionality reduction that consists in applying a modified version of the Fisher Linear Discriminant (FLD) where we only use part of the classified vectors for training. This transformation amounts to resolving the following estimator:

$$\max_{A \in M_{P \times Q}} \frac{A^t S_t A}{A^t S_w A} \tag{1}$$

transforming from a space with P dimensions to a space with Q dimensions where A is the searched transformation matrix, $M_{P \times Q}$ is the space of all transformations, S_t is the total covariance matrix and S_w is the inner-class covariance matrix.

The second part consists in applying the actual RCA transformation, which scales down those dimensions that have great variability within our classes by whitening the resulting feature space. We first calculate the covariance for all the centered data-points in the chunklets:

$$\hat{C} = \frac{1}{p} \sum_{j=1}^{k} \sum_{i=1}^{n_j} (x_{ji} - \overline{x}_j)(x_{ji} - \overline{x}_j)^t \tag{2}$$

where p is the total number of points in the chunklets and $\overline{x}_j$ is the mean of the data-points of the chunklet j. Finally we obtain the whitening matrix:

$$W = \hat{C}^{-\frac{1}{2}} \tag{3}$$

so the new feature space is given by:

$$x_{new} = Wx \tag{4}$$

Our classification algorithm is made of a K-nn classifier using a weighted distance based on two distances. One is from the reduced space mentioned previously where we use the euclidean distance. The other is the Kullback-Leibler distance applied to MFCCs.

$$Dist = \alpha(KL_{MFCC}) + (1 - \alpha)(Euclidean_{RCA}) \tag{5}$$

We optimize the weight α between both distances with a cross-validation technique on the training set.

3. EVALUATION

In this part we discuss the evaluation results of each algorithm and compare them with the other submissions (Waiting for the results....)

[TO BE UPDATED]

4. ACKNOWLEDGMENTS

We want to thank the people from the Music Technology Group (Universitat Pompeu Fabra, Barcelona) and especially those who contributed to Essentia and Gaia, particularly Eduard Aylon.

5. REFERENCES

[1] N. Wack, E. Guaus, C. Laurier, O. Meyers, R. Marxer, D. Bogdanov, J. Serr, P. Herrera. (2009). Music type groupers (MTG): generic music classification algorithms. *Music Information Retrieval Evaluation eXchange (MIREX) extended abstract.*

[2] Essentia & Gaia: audio analysis and music matching C++ libraries developed by the MTG (Resp.: Nicolas Wack), http://mtg.upf.edu/technologies/essentia

[3] X. Hu, S.J. Downie, C. Laurier, M. Bay, A.F. Ehmann (2008). The 2007 MIREX Audio Mood Classification Task: Lessons Learned. 9th International Conference on Music Information Retrieval.

[4] B.E. Boser, I.M. Guyon, V.N. Vapnik. A training algorithm for optimal margin classifiers. *In COLT '92: Proceedings of the fifth annual workshop on Computational learning theory, (pp. 144-152).* New York, NY, USA: ACM, 1992.

[5] C. Laurier, P. Herrera. Audio music mood classification using support vector machine. *Music Information Retrieval Evaluation eXchange (MIREX) extended abstract*, 2007.

[6] C.C. Chang, C.J. Lin. LIBSVM: a library for support vector machines, 2001. *Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm*