Feature Selection in a Cartesian Ensemble of Feature Subspace Classifiers for Music Categorisation

Rudolf Mayer, Andreas Rauber
Vienna University of Technology, Austria
Department of Software Technology & Interactive Systems

Pedro. J. Ponce de León,
Carlos Pérez-Sancho, José M. Iñesta
University of Alicante, Spain
Department of Software & Computing Systems

ABSTRACT

We evaluate the impact of feature selection on the classification accuracy and the achieved dimensionality reduction, which benefits the time needed on training classification models. Our classification scheme therein is a *Cartesian ensemble* classification system, based on the principle of *late fusion* and feature subspaces. These feature subspaces describe different aspects of the same data set. We use it for the ensemble classification of multiple feature sets from the audio and symbolic domains. We present an extensive set of experiments in the context of music genre classification, based on Music IR benchmark datasets. We show that while feature selection does not benefit classification accuracy, it greatly reduces the dimensionality of each feature subspace, and thus adds to great gains in the time needed to train the individual classification models that form the ensemble.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: [retrieval models, search process, selection process]

General Terms

Algorithms, Measurement, Experimentation, Performance

Keywords

Musical genre classification, ensemble classification, feature selection, feature reduction

1. INTRODUCTION

Classification of music into different categories is an important task for retrieval and organisation of music libraries. Previous studies reported a glass ceiling reached using timbral audio features for music classification [1]. We recently presented an approach that is based on the assumption that a diversity of music descriptors and machine learning algorithms are able to make further improvements [8]. We therein created a *Cartesian ensemble* system with these two dimensions (feature sets, learning schemes) as input and train models for each combination of those two input dimensions. In a set of experiments on datasets widely used for musical genre classification, we showed the usefulness of this ensemble approach, in terms of a higher classification accuracy achieved. We further demonstrated that, with the ensemble approach, the user is liberated from the task of a-priori selecting the best set of feature subspaces, and the best classification algorithm.

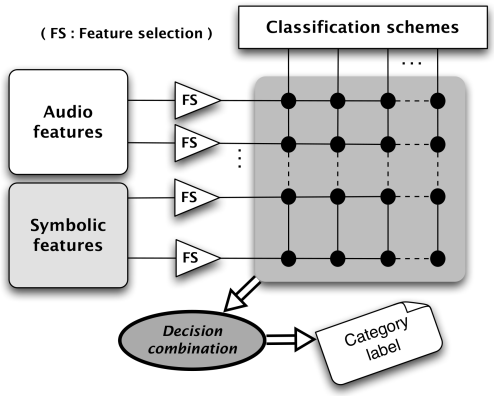

Figure 1: Architecture of the cartesian ensemble

While this approach is shown to successfully reducing the need for (expert) user choices, and yields better classification, the time needed to train the classification models increases significantly, as when compared with just one single classifier on a single feature set. Therefore, in this paper we evaluate the effect of feature selection in our ensemble approach. We want to investigate whether feature selection can be utilised to achieve the same, or at most a slightly worse classification accuracy than with full feature sets, but at the same time whether the feature reduction effect can help to reduce the time needed for training the ensemble.

Section 2 gives a brief overview on related work in the area of ensemble learning. Section 3 introduces our ensemble system In Section 4, we evaluate the system on the task of musical genre classification, Finally, Section 5 provides conclusions and an outlook on future work.

2. RELATED WORK

The Autonomous Classification Engine ACE [11] is a gen-

MML'10, October 25, 2010, Firenze, Italy.

Table 1: Summary of weighted combination rules

SWV	Simple Weighted Vote
RSWV	Rescaled Simple Weighted Vote
BWWV	Best-Worst Weighted Vote
QBWWV	Quadratic Best-Worst Weighted Vote
WMV	Weighted Majority Vote

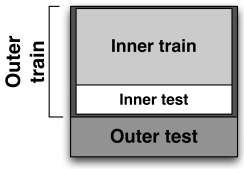

Figure 2: Inner and outer cross-validation scheme.

eral framework for *model selection*, i.e. the task of selecting one classification model from a pool of models. ACE trains a range of classifiers, with different parameters and feature selection methods, and selects the most fitting ones for the current task. The combination of different segments extracted from the same song is studied in [2]. The approach is based on grouping and aggregating non-overlapping blocks of consecutive frames into segments. The segments are then classified individually and the results are aggregated for a song by majority voting. Three different ensemble methods and their applicability to music are investigated in [4]: (1) based on a *one against all* scheme, i.e. for each class, a classifier is trained on the class and its complement, (2) based on building a classifier for each pairwise combination of classes, and (3) by training different classifiers on different feature subspaces. In all methods, the final class label is determined by the probabilities of the individual classifiers.

Our original motivation has been to combine multiple approaches from the music information retrieval (MIR) domain in order to improve (the reliability of) genre classification results based on the assumption that the various music descriptors are complementary. This has been shown by combining different features extracted from the audio signal, namely spectrum-based audio features that cover timbral and rhythmic aspects of the sound, with symbolic descriptors, based on note and chord sequence statistics [9]. A similar multi-modal approach was taken in [10], where the audio features were combined with descriptors extracted from the textual content of the lyrics of the songs. Both studies report gains in genre classification accuracy when simply concatenating the descriptors, i.e. feature fusion.

3. CARTESIAN ENSEMBLE SYSTEM

The system depicted in Figure 1 was first introduced in [8], and builds on *late fusion*. It is called a *Cartesian ensemble* since the set of models it uses as base classifiers is the cartesian product of D feature subspaces by C classification schemes (a specific algorithm with specific algorithm parameters, if any). Each model is built by training classification scheme c_i on feature subspace d_j.

The primary aim is to obtain a *diverse* ensemble of models that will, up to a certain degree, guarantee an improvement of the ensemble accuracy over the best single model trained. Secondly, the ensemble liberates the analyst from the need to select a particular combination of classification scheme and feature subspace to use. A constraint for the system is that the ensemble has to provide results that are at least comparable to the best single scheme. Experimental evaluation has shown that this constraint can be fulfilled.

Pareto-optimal Classifier Selection. Model diversity is a key design factor for building effective classifier ensembles [7]. This has been empirically shown to improve the accuracy of an ensemble over its base models.

For selecting the most diverse models within the ensemble, the *Pareto-optimal* selection strategy is applied in order to discard models not diverse or not accurate enough. The strategy is based on finding the Pareto-optimal set of models by rating them in pairs, according to two measures [7]. The first one is the **inter-rater agreement** diversity measure κ, defined on the coincidence matrix M of the two models. The matrix element $m_{r,s}$ is the proportion of the dataset, which model h_i labels as L_r and model h_j labels as L_s. The second is the **pair average error**. The Pareto-optimal set contains all non-dominated pairs. A pair of classifiers is non-dominated iff there is no other pair that is better than it on both criteria.

Combination Rules. When a new music instance is presented to the ensemble, predictions are made by each of the models. They are then combined, to produce a single category prediction. A number of decision *combination rules* can be used for this final prediction. Our system provides both weighted and unweighted voting rules.

Unweighted rules, described e.g. in [6], include e.g. simple majority voting (MAJ), which favours the class predicted by most votes, or combine the individual results by the average (AVG), median (MED) or max (MAX) of the posterior probability $P(L_k|\mathbf{x}_i)$ of instance x to belong to category L_k, as provided by model h_i.

Weighted rules multiply model decisions by weights and select the label L_k that gets the maximum score. Model weights are based on the estimated accuracy α_i of trained models. The *authority* a_i of each model h_i is established as a function of α_i, normalised, and used as its weight ω_i. Table 1 gives an overview on the weighted rules used in our system. WMV is a theoretically optimal weighted vote rule described in [7], where model weights are set proportionally to $\log(\alpha_i/(1-\alpha_i))$. For more details, especially on the weight functions, please refer to [12, 8].

Inner/Outer Cross Validation. Cross validation is a well-known technique for assessing how the results from a classifier will generalize on independent data. To reduce variability, multiple rounds of partitioning the data in a training and validation (or test) are performed. For weighted combination rules, we also need to estimate the accuracy of individual ensemble models (α_i). In order to avoid using test data of the ensemble for single model accuracy estimation, an *inner cross-validation* relying only on ensemble training data is performed, as depicted in Figure 2.

3.1 Feature selection

In this work a feature selection stage was added to the system presented in [8]. The feature selection method used is a *fast correlation-based filter (FCBF)* described in [16]. It is a feature search method that uses a *symmetrical uncertainty (SU)* correlation-based measure to evaluate features. This measure indicates how much of a feature can be predicted given the information in another feature. The method finds

Table 2: Datasets used in experiments

Dataset	files	genres	file length	ref.
9GDB	856	9	full	[13]
GTZAN	1000	10	30 sec	[15]
ISMIRgenre	1458	6	full	[5]
ISMIRrhythm	698	8	30 sec	[5]

a set of *predominant* features in two steps. First, *relevant* features are ranked according to their SU value with respect to the class (SU_c). A threshold δ on SU can be established to discard features relevant not enough. In the second step, redundant features are further discarded. A feature F_q with rank q is considered redundant if its SU with respect to any feature F_p such that $p < q$ is greater than its SU_c. FCBF has been shown to efficiently achieve high degree of dimensionality reduction for high-dimensional data, while enhancing or maintaining predictive accuracy with selected features.

4. EVALUATION

An overview on the dataset used is given in Table 2; either full songs or 30 second excerpts were available.

4.1 Music Descriptors

We use two sources of input to our ensemble music classification approach: audio features extracted from the waveform, and symbolic descriptors derived from MIDI files, which are obtained through a transcription system. We employ features that proved well in our previous works [3, 9, 8]. However, arbitrary feature sets can be used. The number of features in each subspace is shown in Table 3.

Audio Features. All audio descriptors are extracted from a spectral representation of the audio signal, partitioned into segments. Features are extracted segment-wise, and then aggregated for a piece of music using the median or mean.

The computation of **Rhythm Patterns** is composed of two stages. First, the specific loudness sensation on the 24 critical frequency bands of the Bark scale is computed, resulting in a psycho-acoustical representation reflecting human loudness sensation. Secondly, a spectrum of loudness amplitude modulation per modulation frequency for the bands is computed [9].

A **Rhythm Histogram** (RH) aggregates the modulation amplitude values of the critical bands computed in a Rhythm Pattern and is a descriptor for general rhythmic characteristics in a piece of audio [9]. **Statistical Spectrum Descriptors** (SSD) compute, at the end of the first stage of RPs, a set of statistical values for each critical band, capturing both timbral and rhythmic information very well [9]. **Modulation Frequency Variance Descriptors** (MVD) measure variations in the critical bands for a specific modulation frequency of the Rhythm Pattern matrix, by taking statistics for each modulation frequency over the bands [8].

To incorporate time series aspects, we introduced **TRH** and **TSSD** features, describing variations over time in RH and SSD descriptors, resp. TRH thus captures change and variation of rhythmic aspects in time, and TSSD reflects a change of rhythmics, instruments, voices, etc. over SSD [8].

Symbolic Features. Each audio sample is transcribed to a polyphonic stream of notes in MIDI format by means of the algorithm described in [14]. A set of statistical descriptors is then extracted from transcribed notes as in [3]. Note pitches, pitch intervals, note durations, silence durations, Inter Onset Intervals (IOI) and non-diatonic notes are described by min/max values, range, average, standard deviation, and a normality distribution estimator. Other features include overall statistics such as the average number of notes per beat, or the number of syncopations in the song.

Most of these features are somewhat 'melody-oriented' (e.g. interval-based features). In order to capture relevant information about the polyphonic structure, a distribution of common chord types is computed.

4.2 Classification Schemes and Parameters

For our experiments, we set the system to perform 10-fold outer cross-validation and 3-fold inner cross-validation, and build all subspace models. As for the classification schemes, a selection of classifiers from the *Weka* toolkit has been made, aiming at choosing schemes from different machine learning paradigms, as done in [8].

4.3 Feature selection

Feature selection is performed independently for each subspace at each fold, using outer training data. Table 4 presents a summary of the best single model classification results. The fourth column shows the average number of features selected per fold, also as a percentage of the whole feature subspace. The right section of the table shows performance results without feature selection. This comparison suggests that SVM classifiers' performance on these datasets deteriorates more than schemes based on decision trees.

Table 3 provides further insight on the feature selection step on the GTZAN corpus. These results are averaged over cross-validation folds. The third column shows that the greatest dimensionality reduction is obtained for large audio feature subspaces. The fourth column indicates how many features have been selected at least once. It is worth noting that for the larger subspace, only 3.4% of the features have been selected once. The next column shows how many features are always selected. Here, the SSD subspace shows the best ratio of very predominant features. The accuracy column indicates average accuracy for the best single model trained. The Random Forest scheme shows best performance most of the time.

Feature selection times are negligible when compared to training times, as shown in Table 5. Moreover, training and testing times are an order of magnitude lower than those obtained using the ensemble with all features available. Though, in general, the accuracy of single models decreases when applying a feature selection step, it remains reasonably good for most of the benchmarking corpus.

Table 5: Ensemble cross-validation execution times (in seconds) with and without feature selection (test times are averaged over combinations methods).

	all features		with feature selection		
Corpus	train	test	train	feat. sel.	test
9GDB	6645	140	905	93	18
GTZAN	10702	345	1247	96	23
ISMIRgenre	12510	275	1244	133	23
ISMIRrhythm	5466	185	707	60	8

Table 3: Feature selection results for the GTZAN corpus.

Feature subspace	# feats.	avg. selected	at least once	always	best acc (%)
RH	60	2 (3.3%)	7 (11.7%)	0	27.5 (NB)
RP	1440	9.1 (0.6%)	49 (3.4%)	1	41.2 (RF)
SSD	168	7.4 (4.4%)	15 (8.9%)	4	49.3 (RF)
TRH	420	2 (0.5%)	7 (1.7%)	0	28.1 (NB)
TSSD	1176	19.1 (1.6%)	71 (6%)	4	53.1 (RF)
MVD	420	4.6 (1.1%)	22 (5.2%)	1	31 (RF)
SYMB	63	7.3 (11.8%)	14 (22.6%)	3	44.2 (SVM-quad)

Table 4: Best results on single subspace/classifier combinations on different datasets

Dataset	Classifier	Subspace	avg. feat. sel.	Acc. (%)	Acc. (%) all feat.	Best Acc. all feat.
9GDB	RandomForest	TSSD	42.6 (3.6%)	73.2	76.5	78.2 (SVM-Puk/TSSD)
GTZAN	RandomForest	TSSD	19.1 (1.6%)	53.1	58.2	72.6 (SVM-lin/SSD)
ISMIRgenre	RandomForest	TSSD	20.5 (1.7%)	72.4	73.2	81.3 (SVM-quad/TSSD)
ISMIRrhythm	SVM-lin	RP	38.0 (2.6%)	81.6	88.0	88.0 (SVM-lin/RP)

Table 6: Results of the ensemble classification

Rule	9GDB	GTZAN	ISMIR genre	ISMIR rhythm
Single best	73.2	53.1	72.4	81.6
MAJ	71.38	61	55.83	81.09
MAX	63.9	45.9	67.97	59.46
MED	64.72	42.8	54.39	66.19
AVG	**80.02**	**65.2**	70.23	83.52
SWV	77.69	61.7	56.1	82.95
RSWV	78.27	62	57.96	83.81
BWWV	78.97	61.8	67.08	83.67
QBWWV	79.67	60.7	71.81	**84.53**
WMV	77.92	53.2	**72.43**	84.24
Best wo/ FS	81.66	77.50	84.02	89.11

5. CONCLUSIONS

We presented a classification system based on an ensemble models built on feature subspaces that describe different aspects of a given corpus. the system integrates a feature selection stage, aimed at speeding up training and testing phases, while maintaining a good accuracy level on the task of music genre classification. Our concern in this work was to integrate a feature selection step in the cartesian ensemble and evaluate its impact on the ensemble performance on several datasets. The size of feature subspaces has been greatly reduced to less than 4% of their original size, on average. This has been achieved while maintaining the classification accuracy at a reasonable good level at least for two of the benchmarking corpora, 9GDB and ISMIRrhythm, as shown in Table 6.

Acknowledgments

This work is supported by the Spanish Ministry projects: TIN2009-14247-C02-02, Consolider Ingenio 2010 (MIPRCV CSD2007-00018), and Spain-Austria action (HU2007-25).

6. REFERENCES

[1] J.-J. Aucouturier and F. Pachet. Improving timbre similarity: How high is the sky? *Journal of Negative Results in Speech and Audio Sciences*, 1(1), 2004.

[2] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, and B. Kegl. Aggregate features and Adaboost for music classification. *Machine Learning*, 65:473–484, 2006.

[3] P. J. P. de León and J. M. Iñesta. A pattern recognition approach for music style identification using shallow statistical descriptors. *IEEE Trans. on Systems Man and Cybernetics C*, 37(2):248–257, 2007.

[4] M. Grimaldi, P. Cunningham, and A. Kokaram. An evaluation of alternative feature selection strategies and ensemble techniques for classifying music. In *Proc. Workshop on Multimedia Discovery and Mining*, 2003.

[5] ISMIR 2004 Audio Description Contest. `http://ismir2004.ismir.net/ISMIR_Contest.html`.

[6] J. Kittler, M. Hatef, R. P. Duin, and J. Matas. On combining classifiers. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(3):226–239, 1998.

[7] L. I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004.

[8] T. Lidy, R. Mayer, A. Rauber, P. J. P. de León, A. Pertusa, and J. M. Iñesta. A cartesian ensemble of feature subspace classifiers for music categorization. In *Proc. ISMIR*, Utrecht, Aug. 2010.

[9] T. Lidy, A. Rauber, A. Pertusa, and J. Iñesta. Improving genre classification by combination of audio and symbolic descriptors using a transcription system. In *Proc. ISMIR*, Vienna, Austria, 2007.

[10] R. Mayer, R. Neumayer, and A. Rauber. Combination of audio and lyrics features for genre classification in digital audio collections. In *Proc. ACM Multimedia 2008*, pages 159–168, October 27-31 2008.

[11] C. McKay, R. Fiebrink, D. McEnnis, B. Li, and I. Fujinaga. Ace: A framework for optimizing music classification. In *Proc. ISMIR*, London, UK, 2005.

[12] F. Moreno-Seco, J. M. Iñesta, P. P. de León, and L. Micó. Comparison of classifier fusion methods for classification in pattern recognition tasks. *LNCS*, 4109:705–713, 2006.

[13] C. Perez-Sancho, D. Rizo, and J. M. Iñesta. Genre classification using chords and stochastic language models. *Connection Science*, 21(2 & 3):145–159, 2009.

[14] A. Pertusa and J. M. Iñesta. Multiple fundamental frequency estimation using Gaussian smoothness. In *Proc. IEEE ICASSP*, Las Vegas, USA, 2008.

[15] G. Tzanetakis. *Manipulation, Analysis and Retrieval Systems for Audio Signals*. PhD thesis, Princeton University, 2002.

[16] L. Yu and H. Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proc. ICML*, pages 856–863, 2003.