EVALUATION OF ALBUM EFFECT FOR FEATURE SELECTION IN MUSIC GENRE RECOGNITION

Igor Vatolkin
TU Dortmund
Department of Computer Science
igor.vatolkin@udo.edu

Günter Rudolph
TU Dortmund
Department of Computer Science
guenter.rudolph@udo.edu

Claus Weihs
TU Dortmund
Faculty of Statistics
claus.weihs@udo.edu

ABSTRACT

With an increasing number of available music characteristics, feature selection becomes more important for various categorisation tasks, helping to identify relevant features and remove irrelevant and redundant ones. Another advantage is the decrease of runtime and storage demands. However, sometimes feature selection may lead to "over-optimisation" when data in the optimisation set is too different from data in the independent validation set. In this paper, we extend our previous work on feature selection for music genre recognition and focus on so-called "album effect" meaning that optimised classification models may overemphasize relevant characteristics of particular artists and albums rather than learning relevant properties of genres. For that case we examine the performance of classification models on two validation sets after the optimisation with feature selection: the first set with tracks not used for training and feature selection but randomly selected from the same albums, and the second set with tracks selected from other albums. As it can be expected, the classification performance on the second set decreases. Nevertheless, in almost all cases the feature selection remains beneficial compared to complete feature sets and a baseline using MFCCs, if applied for an ensemble of classifiers, proving robust generalisation performance.

1. INTRODUCTION

Among many different scenarios for automatic classification of music data (we refer to [4] for an introduction to content-based music information retrieval and an overview of related tasks), the recognition of high-level music categories such as music genres and styles is one of the most prominent and user-related applications. Probably the first study on automatic categorisation of music was addressed to distinguish between several classical and popular pieces [22]. After the seminal work of Tzanetakis and Cook on classifying musical data into a hierarchy of 25 music genres and speech categories [38] many efforts were spent to enhance the methods, develop new features, and integrate actual techniques from machine learning research [42]. [37] lists several hundreds of studies related only to the recognition of genres. Since 2005, audio genre classification belongs to tasks of the annual MIREX contest [6].

The operating principle of supervised classification is based on two stages: the training of a classification model $\mathcal{CT}$ and its application $\mathcal{C}$ on uncategorised data:

$$\mathcal{CT} : \left(\mathbf{X} \in \mathbb{R}^{F \times T_{TR}}, \mathbf{y}_L \in \mathbb{R}^{T_{TR}}\right) \mapsto \mathcal{M},$$
$$\mathcal{C} : \left(\mathbf{X} \in \mathbb{R}^{F \times T}, \mathcal{M}\right) \mapsto \mathbf{y}_P \in \mathbb{R}^{T}. \quad (1)$$

Given a set of F numeric data characteristics, or features, for T_{TR} data instances (also referred to as classification windows) resulting in the feature matrix $\mathbf{X}$, and the corresponding labels $\mathbf{y}_L$, the training stage identifies relevant dependencies between features and labels and stores them as a model $\mathcal{M}$. Some approaches are based on the estimation of probability-based distribution of features (Naive Bayes) or boundaries between data instances of different categories (support vector machines); for an overview of classification approaches see, e.g., [13, 43]. Once the classification models are saved, they can be applied to classify T unlabelled data instances represented by the same F previously extracted features.

Music classification can be carried out using features from different sources. For instance, the score allows a precise estimation of harmonic, instrumental, and rhythmic descriptors of music pieces, but it is not always available for popular music. Meta data, cultural features, or tags provide another source of information, but are sometimes incomplete or erroneous. Audio features can be extracted for every digitised music piece, and many classification approaches are limited to or focused on this kind of features [9, 18, 19, 21, 27, 34, 36, 38, 40]. Another advantage of these characteristics is that they are not dependent on the popularity of a song, availability of the score, or Internet connection for the download of metadata. Even if audio features typically require high computing efforts for their extraction, these costs can be reduced to a certain degree if the extraction is done offline or on a server farm. In that case only the time for the training and the application of classification models will influence a user's satisfaction during the definition of new categorisation tasks. For these reasons we have limited the scope of this study to audio features only.

Having a large number of available descriptors at hand, individual features may be very important or completely useless depending on a categorisation task. As the combination of features from different sources may increase the classification quality (e.g., as shown for audio, symbolic, and cultural features in [26]), the inclusion of features from many sources would lead to an increased number of both relevant and irrelevant features. If the number of irrelevant features would become too high, the classification quality may suffer because the probability increases that some irrelevant features are identified as relevant by chance [13, 43]. A solution is to start with a sufficiently large initial feature set and to remove irrelevant and noisy characteristics for a current category by means of feature selection (FS). Other benefits of FS are that classification models created with less features often require less storage space, the classification is done faster, and the danger of overfitting towards the training set may be reduced using a proper evaluation of models and feature sets [3].

In our previous work we have applied feature selection for the recognition of music genres and styles and measured a significant increase of classification performance compared to complete feature sets [39]. For the final evaluation of models optimised with feature selection, we used an independent validation set with tracks not used for model training and feature selection. The motivation for an independent evaluation in music classification is discussed in [9]. However, strictly observed, the validation set used in our previous experiments was not completely independent: due to the limited size of our music database, music pieces for validation were different from training and optimisation sets, but randomly selected from the same albums. Therefore, a danger existed that optimised classification models would have an especially high performance on music pieces of the same artists and albums.

Such effect was observed in [30] for the recognition of genres. Also the tags of songs belonging to the same albums may have higher co-occurrences as inspected in [20]. Further investigations showed interesting results on the difference between album and artist effect for music databases of different sizes [10] as well as varying impact of artist filter with regard to music from different geographic locations around the world [15]. However, none of these studies explicitly evaluated the sensitivity of FS to artist/album effect using a large number of features. Such evaluations can be promising in future, in particular because both latter papers stated differences in measured artist effect for different feature groups, even if the overall numbers of integrated features were not very high.

Thus, the idea behind this study was to re-evaluate the measured advantage of feature selection using a new "album-independent" validation set and to estimate the album effect for different music categories. In the next section, we outline basic concepts of feature selection and refer to several applications. Section 3 describes the setup of the study. In Section 4, the results and the album effect on feature selection are discussed. We conclude with a brief summary of the work and outline steps for future research.

2. FEATURE SELECTION

For an exhaustive introduction into feature selection methods see [12]. In general, the task of feature selection is to find an optimal feature subset indicated by the binary vector $\mathbf{q}$ ($q_i = 1$ for the i-th feature to be selected, otherwise $q_i = 0$), so that some relevance function, or evaluation criterion m (e.g., classification error) is minimised. The functions to maximise (e.g., accuracy) can be easily adapted for minimisation. We define the task of feature selection as:

$$\mathbf{q}^* = \arg\min_{\mathbf{q}} \left[m\left(\mathbf{y}_L, \mathbf{y}_P, \Phi(\mathbf{x}, \mathbf{q})\right)\right], \tag{2}$$

where $\Phi(\mathbf{x}, \mathbf{q})$ corresponds to the subset of the original feature vector $\mathbf{x}$. $\mathbf{y}_L \in [0; 1]$ are the labelled category relationships of classification instances, and $\mathbf{y}_P \in [0; 1]$ are the predicted category relationships. Note that in general m may not necessarily depend on labels, e.g., if the correlation between features is used as selection criterion, or if labels are not available (as in unsupervised classification).

Feature selection with regard to only one evaluation criterion may lead to a decrease of performance for other ones. For example, classification models built with too many features may have smaller classification errors for a specific data set, but be slower and have a poor generalisation performance on other data. Therefore, several relevance functions or objectives $m_1, ..., m_O$ may be considered for simultaneous optimisation:

$$\begin{aligned}\mathbf{q}^* = \arg\min_{\mathbf{q}} [&m_1\left(\mathbf{y}_L, \mathbf{y}_P, \Phi(\mathbf{x}, \mathbf{q})\right), ..., \\ &m_O\left(\mathbf{y}_L, \mathbf{y}_P, \Phi(\mathbf{x}, \mathbf{q})\right)].\end{aligned} \tag{3}$$

In literature, individual features are often referred to as relevant or redundant w.r.t. the performance of a Bayesian classifier which predicts labels based on a probabilistic distribution of feature vectors. For a given feature set X, a feature subset $X' \subset X$ is called *relevant*, iff its removal will decrease the performance of a Bayesian classifier:

$$\begin{aligned}&P(y_P|y_L = y_P, X) < P(y_P|y_L = y_P, X \setminus X') \text{ and} \\ &P(y_P|y_L \neq y_P, X) > P(y_P|y_L \neq y_P, X \setminus X').\end{aligned} \tag{4}$$

A *redundant* feature subset X' can be replaced without decrease of a Bayesian classifier's performance by at least one subset S, which does not contain X':

$$\exists S \subseteq X, X' \cap S = \emptyset : P(y_P|X) = P(y_P|S). \tag{5}$$

The equations (4)–(5) can be adapted to any relevance function, describing a decrease of performance after the removal of relevant features and retaining it after the removal of redundant features.

FS is a very complex task: for F features, the number of all possible non-empty feature subsets is $2^F - 1$, and the related problems were described as NP-hard [1,14]. Therefore, metaheuristics like evolutionary algorithms (EA) [33] which simulate the natural evolution based on principles of recombination (keeping the positive characteristics of solutions) and mutation (exploring the search space using

some random procedure) are a possible remedy. EAs have proven their ability to solve many complex optimisation tasks, among others for data mining and classification [28]. The first application of EAs for FS was introduced in [35], and EAs were recommended for sets with more than 100 features in [16] after the comparison of 18 FS methods.

FS has been already often applied for music classification, for example for the recognition of musical instruments (44 features) [5], moods (66 features) [34], or several classification tasks (between 60 and 1140 features) [25]. Evolutionary FS was integrated in music classification for the first time in [11] and was applied also in later studies, e.g., [8, 27, 36]. The first application of evolutionary multi-objective algorithms to FS for the simultaneous minimisation of the number of features and misclassification rate was proposed in [7]. In music classification, multi-objective evolutionary feature selection was introduced in [40] for genre categorisation and later for the recognition of instruments [41].

In the following, we will describe the study to measure the impact of two-objective FS (minimisation of the classification error and the number of features) on the classification into music genres and styles. Classification results are compared to models built with full feature sets and a baseline with MFCCs. Further, we will investigate the sensitivity of the proposed method to the album effect.

3. EXPERIMENTAL SETUP

3.1 Categorisation Tasks

We distinguish between music genres and styles provided by AllMusicGuide, where a track may belong to one music genre and up to several music styles which are more specific and are typically harder to predict.

Our main database for experiments consists of 120 albums with approximately one third of commercial popular music (45 Pop/Rock albums) as well as tracks of several other genres for a better evaluation of generalisation performance (15 albums of each genre Classic, Electronic, Jazz, Rap, and R&B). For the evaluation of the album effect the database was extended with 120 songs from albums of other artists but the same genre and similar style distribution. It is important to mention that we use our own database, because many publicly available ones were not well suited for this work. Several databases contain only segments of songs so that it is not possible to extract features from long frames (e.g., structural complexity, see the next section). Others are strongly biased towards certain genres or are expensive because of a large share of commercial music. These problems could be in principle avoided using data sets with features only (e.g., Echo Nest descriptors). However, a sufficiently large number of audio features is necessary to measure the impact of feature selection, and many descriptors are developed by ourselves being not available in freely distributed feature sets.

We distinguish between training, optimisation, and two test sets (all of them are disjunct on track level, i.e. it is not permitted to have the same track in more than one set). Each classification model is *trained* from 20 tracks, 10 of which belong to the category to predict (positive examples), and 10 do not belong to it (negative examples). These small training sets are motivated by the real-world situation, where a listener would like to omit high efforts for the labelling of ground truth. On the other side, music pieces have strong variations on different levels (instrumentation, vocal segments, harmony, etc.) and we build classification instances from music intervals of 4 s with 2 s overlap, so that 20 tracks contribute to more than 2,000 classification instances. The data set for the identification of relevant features is the *optimisation* set of 120 songs, each of them selected randomly from the 120 albums. The final evaluation of feature sets after feature selection is done either on 120 *test* tracks randomly selected from the original albums (test set TS) or 120 tracks from other artists (test set TSAI). Thus, the overall number of tracks for each classification experiment was equal to 260. The exact lists of tracks are available on our web site [1] .

3.2 Features

Two large audio feature sets are used as baselines to compare them with sets optimised by means of feature selection. For exact definitions and references please see [39]. The third baseline set is built with MFCCs which are often used for music classification [18].

The first large set comprises low-level audio signal descriptors. Such features can be roughly grouped into timbre, rhythmic, and pitch characteristics [38]. We extend this categorisation to 'timbre and energy', 'chroma and harmony', 'temporal and correlation characteristics', and 'rhythm'. Table 1 provides examples of features for different extraction domains and lists numbers of corresponding feature dimensions. Because we estimate the mean and the standard deviation of each feature vector in a classification window, the original number of 318 dimensions leads to 636 features used for the training of categorisation models.

The second set contains semantic audio features which are closely related to music theory and are listed in Table 2. They can be assigned to four main groups according to their properties and the extraction procedure. The first group consists of chroma-related, harmony, and chord characteristics. The second one comprises temporal, rhythmic, and structural characteristics. The third group (instruments, moods, and various high-level characteristics) relates to features estimated with supervised classification models previously optimised as described in [39]. The last group was extracted using the concept of structural complexity [24]. Here, selected interpretable musical characteristics (instrumentation, harmonic properties, etc.) are represented by a vector of base features, and estimated statistics describe the temporal progress of these vectors over large texture frames.

[1] https://ls11-www.cs.uni-dortmund.de/rudolph/mi#music_test_database

Table 1. Low-level audio features

Groups and examples of features	No.
TIMBRE AND ENERGY - TIME DOMAIN	
Linear prediction coefficients, low energy, peak characteristics	17
TIMBRE AND ENERGY - SPECTRAL DOMAIN	
Various spectral characteristics (bandwidth, centroid, etc.), tristimulus, sub-band energy ratio	29
TIMBRE AND ENERGY - CEPSTRAL DOMAIN	
MFCCs, delta MFCCs, CMRARE modulation features [21]	101
TIMBRE AND ENERGY - PHASE DOMAIN	
Angles and distances [27]	2
TIMBRE AND ENERGY - ERB AND BARK DOMAINS	
Bark scale magnitudes, charact. of ERB bands [17]	53
CHROMA AND HARMONY	
Charact. of spectral peaks, fundamental frequency, chroma, chroma DCT-reduced log pitch (CRP) [29]	101
TEMPORAL AND CORRELATION CHARACTERISTICS	
Characteristics of periodicity peaks	3
RHYTHM	
Characteristics of fluctuation patterns [17]	12

Table 2. Semantic audio features

Groups and examples of features	No.
CHROMA AND HARMONY	
Consonance [23], tonal centroid [17], strengths of major and minor keys [17]	129
CHORD STATISTICS	
Number of different chords and chord changes in 10 s, shares of the most frequent chords [39]	5
TEMPO, RHYTHM AND STRUCTURE	
Duration of music piece, estimated number of beat, tatum, and onset events per minute, tempo, segmentation characteristics after [31]	9
INSTRUMENTS	
Identification of guitar, piano, wind, and strings [41]	32
MOODS	
Aggressive, confident, energetic, etc. [39]	64
VARIOUS HIGH-LEVEL CHARACTERISTICS	
Singing characteristics, effects distortion, characteristics of melodic range [32]	128
STRUCTURAL COMPLEXITY	
Chord, harmony, instruments, tempo and rhythm complexity [39]	70

3.3 Algorithms and Evaluation

The exhaustive tuning of classification methods was beyond the scope of this study - however it was important to test the impact of feature selection and the album effect using classifiers with different operating methods. After preliminary studies, we selected four algorithms. Decision tree C4.5 provides interpretable models and already includes internal feature pruning, but is rather slow. Random forest (RF) creates a large number of unpruned trees based on a randomly drawn subset of features. It is often superior to C4.5 w.r.t. classification quality and is faster, but classification models are not the same if trained another time and are not interpretable. Naive Bayes (NB) is very fast and leads to comprehensible models, especially if they are created from interpretable semantic features. On the other side, it is a probabilistic method which treats feature distributions independently from each other, and classification performance is usually lower. Finally, support vector machine (SVM) is in many cases the state-of-the art method, which achieves the best classification results. However, for the best performance it requires parameter tuning, is slower than other methods, and models have a lower interpretability.

The following two criteria are minimised during feature selection. Because of imbalanced distribution of songs in the optimisation and test sets, the balanced relative error m_{BRE} measures classification quality:

$$m_{BRE} = \frac{1}{2}\left(\frac{FN}{TP+FN} + \frac{FP}{TN+FP}\right), \quad (6)$$

where TP is a number of true positives (tracks belonging to a category and predicted as belonging to it), TN is a number of true negatives (tracks not belonging to a category and predicted as not belonging to it), FP is a number of false positives (tracks not belonging to a category and predicted as belonging to it), and FN is a number of false negatives (tracks belonging to a category and predicted as not belonging to it).

The predicted relationships of tracks to categories are estimated by major voting across all corresponding classification windows:

$$y_P(\mathbf{x}_1, ..., \mathbf{x}_{T_S}; j) = \left\lceil \frac{\sum_{i=1}^{T_S} y_P(\mathbf{x}_i)}{T_S} - 0.5 \right\rceil, \quad (7)$$

where T_S is the number of classification instances in the song j and $\mathbf{x}_i$ describes the feature vector of instance i.

The second optimisation criterion is the selected feature rate m_{SFR}:

$$m_{SFR} = \frac{|\Phi(\mathbf{x}, \mathbf{q})|}{|X|}, \quad (8)$$

where $|\Phi(\mathbf{x}, \mathbf{q})|$ is the number of selected features and $|X|$ the number of all features. m_{SFR} is a rough estimator for runtime and storage demands (classification using a model with more features is typically slower), but may also correlate with the generalisation performance of classification models: models built with less features have a lower tendency to be overfitted towards the training set if the optimisation of feature selection is done using an independent song set.

The feature selection method itself is based on a multi-objective evolutionary algorithm SMS-EMOA [2]. The output is the set of non-comparable feature subsets: the first with the largest m_{SFR} and smallest m_{BRE}, and the last with the smallest m_{SFR} and largest m_{BRE} [2]. Because we focus here on the measurement of album effect having regard to classification error, the discussion of results in the next section is based on subsets with the smallest m_{BRE}. These subsets contain smallest errors achieved for as small feature subsets as possible.

[2] As we minimise both m_{SFR} and m_{BRE}, an example of two non-comparable (also referred to as *non-dominated*) subsets is, e.g., a subset with $m_{SFR} = 0.05$, $m_{BRE} = 0.20$ and another one with $m_{SFR} = 0.10$, $m_{BRE} = 0.15$. The first subset is built with less features and the second one has a smaller classification error.

Table 3. Errors of optimised feature sets and comparison to baselines (smaller values are better). For details see the text.

Categorisation tasks	**Data set**	LOW-LEVEL FEATURES			SEMANTIC FEATURES		
		$\widetilde{m}_{BRE}$	Φ_{LL}	Φ_{MFCC}	$\widetilde{m}_{BRE}$	Φ_{SEM}	Φ_{MFCC}
RECOGNITION OF GENRES							
Classic	TS	0.0127	41.91	33.07	0.0137	*37.43*	*35.68*
	TSAI	0.0175	39.95	*32.47*	0.0270	*57.20*	*50.09*
Electronic	TS	0.0928	66.19	48.91	0.1191	59.25	62.78
	TSAI	0.1040	**82.47**	*56.37*	0.1275	56.52	*69.11*
Jazz	TS	0.0497	66.89	47.56	0.0605	69.86	57.89
	TSAI	0.1192	**107.00**	*89.42*	0.1113	49.47	*83.50*
Pop	TS	0.1291	74.71	68.34	0.1270	43.94	67.23
	TSAI	0.1599	41.20	**75.53**	0.1353	27.91	*63.91*
Rap	TS	0.0508	70.26	56.31	0.0650	*76.29*	*72.06*
	TSAI	0.0475	72.74	**88.29**	0.0579	*56.54*	**107.62**
R&B	TS	0.1570	*89.82*	74.09	0.1484	76.85	69.93
	TSAI	0.1337	84.73	*56.29*	0.1486	73.67	*62.57*
RECOGNITION OF STYLES							
AdultContemporary	TS	0.1192	67.31	55.94	0.1344	57.02	63.07
	TSAI	0.1906	64.83	**81.45**	0.1860	64.27	*79.49*
AlbumRock	TS	0.0900	65.41	46.68	0.1066	51.15	55.29
	TSAI	0.1225	61.47	*49.04*	0.1617	50.73	*64.73*
AlternativePopRock	TS	0.1066	70.92	49.67	0.1092	54.19	50.89
	TSAI	0.1746	71.47	**81.40**	0.1818	64.10	**84.76**
ClubDance	TS	0.1551	82.41	74.35	0.1389	55.02	66.59
	TSAI	0.1398	70.25	**54.69**	0.1465	64.65	57.32
HeavyMetal	TS	0.0839	59.25	*92.20*	0.0778	56.25	*85.49*
	TSAI	0.1192	*59.22*	**85.26**	0.0991	55.06	**70.89**
ProgRock	TS	0.1072	64.00	47.43	0.0973	53.52	43.04
	TSAI	0.1780	57.68	*65.56*	0.2039	59.85	*75.10*
SoftRock	TS	0.1104	67.28	45.19	0.1197	53.13	49.00
	TSAI	0.1752	69.91	**65.28**	0.1498	62.39	55.81
Urban	TS	0.1038	76.95	74.67	0.0837	57.06	60.22
	TSAI	0.1541	59.09	58.81	0.1553	51.87	59.27

4. DISCUSSION OF RESULTS

4.1 Table with Results

Table 3 provides the summary of results and is organised as follows. The first column lists categorisation tasks. The second column indicates whether the album-dependent test set TS or album-independent test set TSAI was used for the final validation. Columns 3-5 describe results with low-level features. In the column 3 the "*mean best*" error $\widetilde{m}_{BRE}$ is listed. The *mean* is here calculated across 10 statistical repetitions: because evolutionary FS is based on random decisions, the results are not the same for each run. So the value of $\widetilde{m}_{BRE} = 0.0127$ corresponds to the *expected* best m_{BRE} after the application of FS. The *best* means that we take into account feature subsets with the smallest m_{BRE} and the largest m_{SFR} across compromise solutions identified with a multi-objective selection approach (see the previous section).

Entries in columns 4 and 5 measure the relative reduction of $\widetilde{m}_{BRE}$ compared to complete set of low-level features, Φ_{LL}, and set of MFCCs, Φ_{MFCC}. Smaller values are better. For example, in the first line $\widetilde{m}_{BRE} = 0.0127$ corresponds to 41.91% of the error of the model which uses all low-level descriptors ($m_{BRE} = 0.0303$ [3]). Similarly, $\widetilde{m}_{BRE}$ is reduced to 33.07% of the error of the model built with MFCCs only.

Columns 6-8 contain values of $\widetilde{m}_{BRE}$ for models built with semantic features and the reduction of error compared to full set of semantic features Φ_{SEM} and Φ_{MFCC}.

4.2 Album Effect and Two Cases where Feature Selection Fails

As it could be expected, classification errors increase for most of categories if we switch from the test set TS to TSAI. The advantage of optimised feature subsets compared to baselines (columns 4,5,7,8) is often decreased, but not always. For instance, despite of a larger error for AdultContemporary using TSAI (0.1906 against 0.1192), the advantage of optimised low-level feature subsets compared to the model with all low-level features is slightly increased (64.83 against 67.31, smaller value is better), but not if compared to the model built with MFCCs (81.45 against 55.94).

A more important observation is that in all but two cases optimised models are better than baselines (only two entries in columns 4,5,7,8 are above 100%) which means that feature subsets after FS lead to a robust reduction of error even if finally validated on the test set from inde-

[3] Please note that we use an ensemble of four classifiers and select the best one for each task. Using a complete feature set for the category Classic leads to $m_{BRE} = 0.0303$ if trained with random forest; for example, using naive Bayes leads to $m_{BRE} = 0.0695$, so that the error of the optimised combination "feature subset and classifier" is even stronger reduced if compared to a simple application of naive Bayes together with all low-level features.

pendent artists and albums. The first exception is Jazz (value of 107.00 in the 4th column): here the full Φ_{LL} set ($m_{BRE} = 0.1114$) leads to a slightly smaller error than the optimised set ($m_{BRE} = 0.1192$). This can be explained by the choice of music: in the artist-independent validation song set the category Jazz was represented rather by European Jazz, where the training and optimisation set contained rather American Jazz [4]. Another exception relates to the error of optimised subsets with semantic features compared to MFCCs for Rap (value of 107.62, column 8). This matches well the theoretical reason that MFCCs are particularly successful for the recognition of speech. The smallest error for Rap is achieved using the optimised set with low-level features (and MFCCs belong to this set): $m_{BRE} = 0.0475$.

4.3 A Further Danger for Feature Selection (or Advantage of Ensembles)

In all but two explained situations FS led to smaller errors. However, this statement holds for classification with four methods. Using an ensemble of classifiers makes often sense, and in our previous work we have already observed that there is no "winner" for all categories [39]. To examine whether the feature selection was successful for individual combinations of a classifier and a task we compared the results to baselines by means of Wilcoxon test. If no statistical advantage against a baseline has been observed for at least one of four classifiers, the corresponding entry in Table 3 is marked with an italic font. If the baseline was even better for at least one classifier, the entry is marked with a bold font. Particularly some models with MFCCs seem to provide a better generalisation performance rather than optimised feature subsets. This happens only if test set TSAI is used for the validation. In other words, optimising feature selection with an individual classifier may lead to overfitting–but in our study this case was avoided using an ensemble of several classifiers.

4.4 A Remark on Resources

Beside possible problems for feature selection discussed above, it should not be forgotten that FS provides a strong advantage against large sets of features because it helps to reduce storage and runtime demands. The advantage of smaller feature sets is that the classification is typically faster [5]. When the time expensive feature selection may be run once for each new music category, the automatic classification based on the optimised feature set can be applied on new songs over and over again. It is hard to precisely measure the reduction of computing demands, especially for experiments on different machines. As a rough measure we may estimate the decrease of runtime of the last FS iteration compared to the first iteration (in each iteration, a classification model is trained and validated). As an example, the mean of runtime of the last iteration divided by runtime of the first iteration for the category Classic is 15.08 for the low-level feature set and 12.57 for the semantic set (classification with C4.5), 34.55 and 31.72 (RF), 20.88 and 8.56 (NB), and 12.68 and 10.54 (SVM).

[4] We came to this explanation after the studies were accomplished. The uniform sampling of European and American Jazz tracks for optimisation and validation sets could be a better decision, but in that case it would not be possible to exactly compare the results to [39].

[5] As we could see, a set of MFCCs is also small and is sometimes successful, so the reduction of demands on resources is not very strong here. However, all but one values in columns 5 and 8 are below 100%, and it is probably not the best idea to build classification models with MFCCs only for all possible classification tasks (styles, tags, moods, etc.)

5. CONCLUSIONS AND OUTLOOK

In this work we have examined whether the success of feature selection in music classification suffers from an "album effect", so that the properties of albums and artists rather than of target categories like genres and styles are learned. As it could be expected, the danger of such overfitting exists, and the performance is typically reduced if the validation set is built with tracks of other artists. However, if there are enough available features at hand, and feature selection is applied using an ensemble of classifiers, in all but two cases the optimised subsets helped to build classification models not only with less features, but also with smaller classification errors compared to baselines. These two cases could be theoretically explained and do not detract the general sense of feature selection - but they underline the consequence that any significant achievements in classification domain raise and fall with the design of data sets. A very simple case observed in this study was that the classification models optimised to recognise particularly American Jazz were not best suited to recognise European Jazz. In future we plan to continue our work investigating advantages and dangers of feature selection for music classification. In particular, the application on publicly available data sets is important for a reliable comparison of results. However, this is a hard task which requires compromises, e.g., limiting the set of features only to available Echo Nest descriptors. Further optimisation of algorithm parameters (e.g., larger ensembles, various kernels for SVMs) is another promising direction.

6. REFERENCES

[1] E. Amaldi and V. Kann. On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theoretical Computer Science*, 209(1-2):237–260, 1998.

[2] N. Beume, B. Naujoks, and M. Emmerich. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181(3):1653–1669, 2007.

[3] B. Bischl, O. Mersmann, H. Trautmann, and C. Weihs. Resampling methods for meta-model validation with recommendations for evolutionary computation. *Evolutionary Computation*, 20(2):249–275, 2012.

[4] M. A. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney. Content-based music information retrieval: Current directions and future challenges. *Proceedings of the IEEE*, 96(4):668–696, 2008.

[5] J. D. Deng, C. Simmermacher, and S. Cranefield. A study on feature analysis for musical instrument classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 38(2):429–438, 2008.

[6] J. S. Downie, A. F. Ehmann, M. Bay, and M. C. Jones. The Music Information Retrieval Evaluation eXchange: Some observations and insights. In Z. W. Ras and A. Wieczorkowska, editors, *Advances in Music Information Retrieval*, pages 93–115. Springer, 2010.

[7] C. Emmanouilidis, A. Hunter, and J. MacIntyre. A multiobjective evolutionary setting for feature selection and a commonality-based crossover operator. In *Proc. IEEE Congress on Evolutionary Computation (CEC)*, volume 1, pages 309–316, 2000.

[8] S. Essid, G. Richard, and B. David. Musical instrument recognition by pairwise classification strategies. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(4):1401–1412, 2006.

[9] R. Fiebrink and I. Fujinaga. Feature selection pitfalls and music classification. In *Proc. 7th Int'l Conf. on Music Information Retrieval (ISMIR)*, pages 340–341, 2006.

[10] A. Flexer and D. Schnitzer. Effects of album and artist filters in audio similarity computed for very large music databases. *Computer Music Journal*, 34(3):20–28, 2010.

[11] I. Fujinaga. Machine recognition of timbre using steady-state tone of acoustic musical instruments. In *Proc. Int'l Computer Music Conf. (ICMC)*, pages 207–210, 1998.

[12] I. Guyon, M. Nikravesh, S. Gunn, and L. A. Zadeh, editors. *Feature Extraction. Foundations and Applications*, volume 207 of *Studies in Fuzziness and Soft Computing*. Springer, Berlin Heidelberg, 2006.

[13] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, New York, 2009.

[14] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, 1997.

[15] A. Kruspe, H. Lukashevich, and J. Abeßer. Artist filtering for non-western music classification. In *Proc. 6th Audio Mostly Conference (AM)*, pages 82–86, 2011.

[16] M. Kudo and J. Sklansky. Comparison of algorithms that select features for pattern classifiers. *Pattern Recognition*, 33(1):25–41, 2000.

[17] O. Lartillot. *MIRtoolbox 1.4 User's Manual*. Finnish Centre of Excellence in Interdisciplinary Music Research and Swiss Center for Affective Sciences, 2012. Online resource.

[18] B. Logan. Mel frequency cepstral coefficients for music modeling. In *Proc. 1st Int'l Symp. on Music Information Retrieval (ISMIR)*, 2000.

[19] M. I. Mandel and D. Ellis. Song-level features and support vector machines for music classification. In *Proc. 6th Int'l Conf. on Music Information Retrieval (ISMIR)*, pages 594–599, 2005.

[20] M. I. Mandel, R. Pascanu, D. Eck, Y. Bengio, L. M. Aiello, R. Schifanella, and F. Menczer. Contextual tag inference. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, 7(Suppl.):32, 2011.

[21] R. Martin and A. M. Nagathil. Cepstral modulation ratio regression (CMRARE) parameters for audio signal analysis and classification. In *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 321–324, 2009.

[22] B. Matityaho and M. Furst. Neural network based model for classification of music type. In *Proc. 18th Convention of Electrical and Electronics Engineers in Israel*, pages 4.3.4/1–4.3.4/5, 1995.

[23] M. Mauch and S. Dixon. Approximate note transcription for the improved identification of difficult chords. In *Proc. 11th Int'l Society for Music Information Retrieval Conf. (ISMIR)*, pages 135–140, 2010.

[24] M. Mauch and M. Levy. Structural change on multiple time scales as a correlate of musical complexity. In *Proc. 12th Int'l Society for Music Information Retrieval Conf. (ISMIR)*, pages 489–494, 2011.

[25] R. Mayer, A. Rauber, P. J. Ponce de León, C. Pérez-Sancho, and J. M. Iñesta. Feature selection in a cartesian ensemble of feature subspace classifiers for music categorisation. In *Proc. 3rd Int'l Workshop on Machine Learning and Music (MML)*, pages 53–56, 2010.

[26] C. McKay. *Automatic Music Classification with jMIR*. PhD thesis, McGill University, 2010.

[27] I. Mierswa and K. Morik. Automatic feature extraction for classifying audio data. *Machine Learning Journal*, 58(2-3):127–149, 2005.

[28] A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, and C. A. Coello Coello. A survey of multiobjective evolutionary algorithms for data mining: Part I. *IEEE Transactions on Evolutionary Computation*, 18(1):4–19, 2014.

[29] M. Müller and S. Ewert. Chroma Toolbox: MATLAB implementations for extracting variants of chroma-based audio features. In *Proc. 12th Int'l Conf. on Music Information Retrieval (ISMIR)*, pages 215–220, 2011.

[30] E. Pampalk, A. Flexer, and G. Widmer. Improvements of audio-based music similarity and genre classificaton. In *Proc. 6th Int'l Conf. on Music Information Retrieval (ISMIR)*, pages 628–633, 2005.

[31] J. Paulus and A. Klapuri. Music structure analysis using a probabilistic fitness measure and an integrated musicological model. In *Proc. 9th Int'l Conf. on Music Information Retrieval (ISMIR)*, pages 369–374, 2008.

[32] G. Rötter, I. Vatolkin, and C. Weihs. Computational prediction of high-level descriptors of music personal categories. In B. Lausen, D. van den Poel, and A. Ultsch, editors, *Algorithms from and for Nature and Life*, pages 529–537. Springer, 2013.

[33] G. Rozenberg, T. Bäck, and J. N. Kok, editors. *Handbook of Natural Computing*. Springer, Berlin Heidelberg, 2012.

[34] P. Saari, T. Eerola, and O. Lartillot. Generalizability and simplicity as criteria in feature selection: Application to mood classification in music. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(6):1802–1812, 2011.

[35] W. W. Siedlecki and J. Sklansky. A note on genetic algorithms for large-scale feature selection. *Pattern Recognition Letters*, 10(5):335–347, 1989.

[36] C. N. Silla Jr., A. L. Koerich, and C. A. A. Kaestner. A feature selection approach for automatic music genre classification. *International Journal of Semantic Computing*, 3(2):183–208, 2009.

[37] B. Sturm. A survey of evaluation in music genre recognition. In *Proc. 10th Int'l Workshop on Adaptive Multimedia Retrieval (AMR)*, 2012.

[38] G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, 2002.

[39] I. Vatolkin. *Improving Supervised Music Classication by Means of Multi-Objective Evolutionary Feature Selection*. PhD thesis, Dep. of Computer Science, TU Dortmund, 2013.

[40] I. Vatolkin, M. Preuß, and G. Rudolph. Multi-objective feature selection in music genre and style recognition tasks. In *Proc. 13th Annual Genetic and Evolutionary Computation Conf. (GECCO)*, pages 411–418, 2011.

[41] I. Vatolkin, M. Preuß, G. Rudolph, M. Eichhoff, and C. Weihs. Multi-objective evolutionary feature selection for instrument recognition in polyphonic audio mixtures. *Soft Computing*, 16(12):2027–2047, 2012.

[42] C. Weihs, U. Ligges, F. Mörchen, and D. Müllensiefen. Classification in music research. *Advances in Data Analysis and Classification*, 1(3):255–291, 2007.

[43] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Elsevier, San Francisco, 2005.