Understandable Models Of Music Collections Based On Exhaustive Feature Generation With Temporal Statistics

Fabian Moerchen
Databionic Research Group
Philipps-University Marburg
Hans-Meerwein-Str., Marburg, Germany
fabian.moerchen @siemens.com

Ingo Mierswa
Artificial Intelligence Unit
University of Dortmund
Baroper Str., Dortmund, Germany
ingo.mierswa@uni-dortmund.de

Alfred Ultsch
Databionic Research Group
Philipps-University Marburg
Hans-Meerwein-Str., Marburg, Germany
ultsch@informatik.uni-marburg.de

ABSTRACT

Data mining in large collections of polyphonic music has recently received increasing interest by companies along with the advent of commercial online distribution of music. Important applications include the categorization of songs into genres and the recommendation of songs according to musical similarity and the customer's musical preferences. Modeling genre or timbre of polyphonic music is at the core of these tasks and has been recognized as a difficult problem. Many audio features have been proposed, but they do not provide easily understandable descriptions of music. They do not explain why a genre was chosen or in which way one song is similar to another. We present an approach that combines large scale feature generation with meta learning techniques to obtain meaningful features for musical similarity. We perform exhaustive feature generation based on temporal statistics and train regression models to summarize a subset of these features into a single descriptor of a particular notion of music. Using several such models we produce a concise semantic description of each song. Genre classification models based on these semantic features are shown to be better understandable and almost as accurate as traditional methods.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Algorithms

Keywords

music mining, feature generation, meta learning, logistic regression, genre classification

KDD'06, August 20–23, 2006, Philadelphia, Pennsylvania, USA.

1. INTRODUCTION

The advent of commercial online distribution of music brings up interesting problems that can be tackled with data mining technologies. Many tasks are still performed largely manually, e.g. the categorization of new music into genres or the detailed analysis of music by the Music Genome Project[1]. A (partial) automation of the musical gene extraction could speed up this ongoing endeavor. The recommendation of music to customers can be performed with itemset methods just like for books or other products. This way only well known music is covered, new or less know music is hardly ever recommended. Direct analysis of polyphonic audio data can help to solve these problems [35].

Confronted with music data, data mining encounters a new challenge of scalability. Music databases store millions of records and each item contains up to several million values. The solution to overcome this issue is to extract features from the audio signal which leads to a strong compression of the data set at hand. Many different audio features extracted from polyphonic music have been proposed for different applications in music information retrieval (e.g. [19, 39, 20, 29, 22, 25]). Artist and genre classification or retrieval of similar music can be performed with machine learning methods utilizing these features. The models can be used for the automatic creation of taxonomies on websites or in music recommendation systems.

The basis of most methods is the extraction of *short-term* features describing the audio content of small time windows. The sequence of short-term features is commonly aggregated, e.g., with mean and standard deviation [39], in order to obtain *long-term* features describing several seconds or minutes of music. Recently, authors have started to use the temporal structure of short-term feature for aggregation [30, 20, 3, 25, 21]. The *bag of frames* [43] methods alternatively summarize the short-term features with mixture models or vector quantization [19, 2].

Many authors use features motivated by heuristics on musical structure [39] and psychoacoustic analysis of frequency and modulation of sound [29]. But not all features need to be relevant for a particular task. Further, distance calculations using very high dimensional vectors [29] can be problematic, because these vectors spaces are inherently sparse and tend to be equidistant [1]. Feature selection techniques can be used to optimize the performance and create smaller repre-

[1] `http://www.pandora.com`

sentations [20, 25]. Even learning such representations can be performed [22, 27], this is however not feasible for large scale applications.

Almost all proposed representations of music are, however, hard to understand. The result of applying signal processing and statistical methods can not easily be explained to the common user of music applications. One notable exception is the approach described in [5]. Short-term audio features are mapped to zero or one depending on the membership in genre or artist categories using supervised learning with feed-forward neural nets. The output of each neural net can then be interpreted as the similarity of the short segment to other segments of songs from a genre or artist. These short-term *semantical* features are subsequently summarized with a mixture model, that cannot easily be used to explain the music recommendations made by the system.

Our work can be seen as the combination of the large scale generation of long-term audio features in [25] with the semantical modeling of [5]. We use logistic regression [16] in order to obtain concise and interpretable features summarizing a subset of the complicated features generated directly from polyphonic audio. Each resulting feature describes the probability of a *complete song* belonging to a certain group of similar music. In comparison to [25] we better utilize the power of the large scale feature generation, because more features are used. The dimensionality of the final representation is kept low through of the summarization by the regression models. Additionally, each feature of this small feature set corresponds to a group of songs. This enables users to easily understand these semantic models compared to models learned from short-term or long-term features alone.

First, some related work is discussed in Section 2 in order to motivate our approach. The large scale audio feature generation is explained in Section 3. The methods we propose for semantic modeling of musical similarity are described in Section 4. The results are presented in Section 5 and discussed in Section 6. A summary is given in Section 7.

2. RELATED WORK AND MOTIVATION

Machine Learning has shown its benefits in many applications on music data [46, 11]. Since many machine learning methods also rely on a good similarity measure between instances, the success of these methods also depends on the quality of the feature sets.

Musical similarity can be modeled using a set of short-term Mel Frequency Cepstral Coefficient (MFCC, e.g. [33]) vectors summarized with a so-called *bag of frames* [43], i.e. the result of a vector quantization method or Gaussian mixture models [19, 2, 43]. This representation make distance calculations between songs problematic. Comparing the Gaussian mixture models of two songs requires calculation of the pairwise likelihood that each song was generated by the other song's model. This representation cannot easily be used with machine learning algorithms that require the calculation of a centroid. It also scales badly with the number of songs, because the pairwise similarities of all songs need to be stored [4].

The seminal work of Tzanetakis [40, 39] is the foundation for many musical genre classification methods. A single feature vector is used to describe a song, opening the problem for many standard machine learning methods. Many follow-ups of this approach tried to improve it by using different features and/or different classifiers. For example wavelet based features with Support Vector Machines (SVM) and Linear Discriminant Analysis [18] or linear predictive coefficients (LPC) and SVM [45].

In [29] several high-dimensional vector feature sets were compared to bag of frames representations measuring the ratio of inner to inter class distances of genres, artists, and albums. The vector-based representation with Spectrum Histogram performed best.

The above methods all rely on general purpose descriptions of music. The ground truth of genre or timbre categories was not used in the construction of the feature sets, except maybe as guidelines for the heuristics used in the feature design and selection of parameters. In contrast, timbre similarity was modeled in [25] by selecting only few features of a large candidate set based on the ground truth of a manually labeled music collection. The timbre features outperformed existing general purpose features on several independent music collections.

Most audio features are extracted from polyphonic audio data by a sequence of processing steps involving sophisticated signal processing and statistical methods. But only few like *beats per minute* are understandable to the typical music listener. Much effort has been put into developing highly specialized methods using musical and psychological background knowledge to derive semantic descriptions e.g. of rhythm, harmony, instrumentation, or intensity (see [13] for a summary). The results are, however, often only understandable to musical experts. The calculation of musical similarity by combining the heterogeneous descriptions for each song is further challenging in itself.

In [5] short-term MFCC features are mapped to more abstract features describing the similarity to a certain genre or artist. This way, short segments of a song can be described by saying that they *sound like country* with a certain probability. The vectors of semantical short term features of a complete song are summarized with mixture models, however, partly destroying the understandability of the results.

We combine the exhaustive generation of long-term audio features [25] with the semantical modeling of [5] to generate interpretable features each describing the probability of a *complete song* to belong to a certain group of music.

Using the predictions of several such learned models in order to derive a final decision is known as ensemble learning [7]. Our approach is loosely related to stacking [44]. Stacking learns the same concept on different subsamples of the data set. Then, the predictions of the learned models build a new feature set which is used to learn a final decision model. In contrast, we learn different concepts on the same sample. For each concept a possibly different feature set is selected and aggregated.

3. AUDIO FEATURE GENERATION

The raw audio data of polyphonic music is not suited for direct analysis with data mining algorithms. It contains various sound impressions that are overlayed in a single (or a few correlated) time series. These time series cannot be compared directly in a meaningful way. The sound of polyphonic music is commonly described by extracting audio features on short time windows during which the sound is assumed to be stationary. We call these descriptors *short-term* features. The down sampled time series of short-term feature values can be aggregated to form so-called *long-term* features describing the music. We introduced many vari-

Table 1: Music collections. For *ISMIR04* the group Jazz also contains Blues and Rock also contains Pop

Genre	GTZAN	ISMIR	MAB	RADIO
Alternative			145	
Blues	100		120	
Classical	100	640		
Country	100			206
Dance	100			204
Electronic		229	113	
Folk			222	
Funk			47	
Hiphop	100		300	208
Jazz	100	52	319	201
Metal	100	90		206
Pop	100		116	
Reggae	100			
Rock	100	203	504	
Soul				205
World		224		201
size	1000	1438	1567	1431

ants of existing short-term features and the consistent use of temporal statistics for long-term features in [25]. The cross-product of short- and long-term functions leads to a large amount of audio features describing various aspects of the sound that we generated with the publically available MusicMiner[26][2] software.

We used four disjoint data sets for the evaluation of our method. The *GTZAN* collection was first used in [39] for classification of musical genre. The *ISMIR04* corpus was used in the ISMIR'04 genre classification contest[3]. The Musical Audio Benchmark (*MAB*) [14][4] data was collected from `www.garageband.com`. Finally, we collected songs from internet *RADIO* stations listed on `www.shoutcast.com` choosing seven distinct genres. The collections are summarized in Table 1.

The audio data was reduced to mono and a sampling frequency of 22kHz. To reduce processing time and avoid lead in and lead out effects, a 30s segment from the center of each song was extracted. For MAB only 10s were available and for GTZAN the given 30s segment was used. The window size was 23ms (512 samples) with 50% overlap. Thus for each short-term feature, a time series with 2582 time points at a sampling rate of 86Hz was produced.

We used the short-term features listed in Table 2. For more details on the features please refer to the original publications listed or [26]. Including some variants obtained by preprocessing the features, e.g., the logarithm of the Chroma features, a total of 140 short-term features was generated.

The long-term features are listed in Table 3. The most simple static aggregations are the empirical moments of the probability distribution of the feature values. We used the first four moments, robust variants by removing the largest and smallest 2.5% of the data prior to estimation, the median, and the median absolute deviation (MAD). These ten statistics are also applied to the first and second order differences and the first and second order absolute differences, generating 40 additional features (Δ and Δ^2 moments).

[2] `http://musicminer.sf.net`
[3] `ismir2004.ismir.net/genre_contest/index.htm`
[4] `http://www-ai.cs.uni-dortmund.de/audio.html`

Table 2: Short-term features

Name	Features
Volume [17]	2
Zerocrossing [17]	2
Lowenergy [39]	2
SpectralCentroid [17]	2
SpectralBandwidth [17]	2
BandEnergyRatio [17]	2
SpectralRolloff [17]	2
SpectralCrestFactor [17]	2
SpectralFlatnessMeasure [17]	2
SpectralSlope [22]	2
SpectralYIntercept [22]	2
SpectralError [22]	2
Mel Magnitudes [33]	34
MFCC [33]	34
Chroma [10]	48
total	140

The first 10 values of the autocorrelation function and slope, intercept, and error of a linear regression of the autocorrelation are used to capture the correlation structure. The spectral centroid and bandwidth as well as the same three regression parameters as above are used to describe the spectrum of the short-term feature time series. Similar to the short-term MFCC, the first 10 cepstrum coefficients of the short-term feature time series are also extracted.

As in [20] the modulation energy was measured in three frequency bands: *"1-2Hz (on the order of musical beat rates), 3-15Hz (on the order of speech syllabic rates) and 20-43Hz (in the lower range of modulations contributing to perceptual roughness)"*. The absolute values were complemented by the relative strengths obtained by dividing each through the sum of all three.

Non-linear analysis of time series [15] offers an alternative way of describing temporal structure that is complementary to the analysis of linear correlation and spectral properties. Similar to the raw audio processing in [22] the reconstructed phase space [36] is used with an embedding dimension of two and time lags 1-10 to obtain a 2-dimensional time series from the univariate short-term features. The moments of the distances and angles in this phase space representation generate a total of 200 long-term feature functions.

The crossproduct of short- and long-term feature functions amounts to $140 \times 284 = 39,760$ long-term audio features[5]. The framework is easily capable of producing several hundred thousand features by activating more short- and long-term modules. Obviously, this can take a lot of computation time and memory. The above feature set requires

[5] The complete list of features can be obtained by emailing the first author.

Table 3: Long-term feature functions

Functions	Features
Moments: $mean(\cdot)$, $std(\cdot)$, $skew(\cdot)$, $kurt(\cdot)$, $mean_{5\%}(\cdot)$, $std_{5\%}(\cdot)$, $skew_{5\%}(\cdot)$, $kurt_{5\%}(\cdot)$, $median(\cdot)$, $mad(\cdot)$	10
Differences: $\{\Delta(\cdot), abs(\Delta(\cdot)), \Delta^2(\cdot), abs(\Delta^2(\cdot))\} \times$ moments	40
Autocorrelation: $ac_1(\cdot), ..., ac_{10}(\cdot)$, $slope(ac(\cdot))$, $yint(ac(\cdot))$, $regerr(ac(\cdot))$	13
Spectrum: $centroid(\cdot)$, $bandwidth(\cdot)$, $slope(\cdot)$, $yint(\cdot)$, $regerr(\cdot)$	5
Cepstrum: $cepstrum_1(\cdot)$, ..., $cepstrum_{10}(\cdot)$	10
Modulation: $mod_{1-2}(\cdot)$, $mod_{3-15}(\cdot)$, $mod_{20-43}(\cdot)$, $nmod_{1-2}(\cdot))$, ..., $nmod_{20-43}(\cdot)$	6
Phasespace: $\{ PS_{2,1}(\cdot), ..., PS_{2,10}(\cdot) \} \times \{ angles(\cdot), dists(\cdot) \} \times$ moments	200
total	284

a reasonable 115 seconds per song on a 2.6GHz system. We also considered an extended feature set. We added variants of the MFCC short-term features using different frequency scales (Bark [47], Equivalent Rectangular Bandwidth (ERB) [23], and Octave) and different orthonormal decompositions (Discrete Cosine Transform and Haar wavelet decomposition). Additional long-term features describe the temporal structure of distances and angles in the phase space. The resulting 688.000 values per song required 40 minutes per song. This made experiments with a large number of songs infeasible with our current resources.

4. SEMANTIC AUDIO FEATURES

In the last section we discussed how each song is described with about 40,000 features. Of course it would be possible to directly use these features in order to learn a classification model which separates the given songs according to the ground truth at hand. However, there are two drawbacks: first, using the complete feature set will cause the usual problems of classification in such high dimensional space, namely curse of dimensionality and higher run times. Second, the short-term and long-term features are rather technical and derived from signal processing, psychoacoustic, and time series analysis techniques. Models learned from up to 40,000 of these complicated features can hardly be understood by end users.

The goal is to simplify the feature set by aggregating the relevant features from the exhaustive feature set into new concise and powerful features. Therefore, we adapt a meta learning idea known as stacking [44]. In contrast to Stacking we do not learn the same concept on different subsamples but different concepts on the same sample.

Let D be the data set describing these different concepts. D is called *ground truth* since the feature aggregation process relies on the quality of the concepts described by this data set. The concepts which should be learned are defined by a partition of the data set into classes, i. e. $D_1 \dots D_K$ such that $D_k \cap D_l \neq \emptyset \Rightarrow D_k = D_l$ and $D = \bigcup_{k=1}^{K} D_k$. Note, that each data point $d \in D$ corresponds to a song represented by the 40,000 features discussed in the previous section.

We can now define K learning tasks based on the classes D_k. For each k we try to separate D_k from $D \setminus D_k$. We use *Bayesian logistic regression* in order to train models for these K classification tasks. The predictions of this learning scheme can directly be interpreted as the likelihood that a given example belongs to the learned class. Since the values are already normalized, it is not necessary to apply post-processing scaling schemes after learning a classification function.

Using Laplace priors for the influence of each feature leads to a built-in feature selection that reduces runtime and avoids over-fitting of the final model. In comparison with Gaussian priors, the Laplace has more weight closer to zero. Irrelevant features are more likely to have final weights of exactly zero excluding them from the model. This corresponds to *"a prior belief that a small portion of the variables have a substantial effect on the outcome while most of the others are most likely unimportant"* [9] and is equivalent [9] to the lasso method [37, 12]. We used the BBR[9][6] software with Laplace priors and auto selection of the parameter λ.

We applied a robust z-transformation to each long-term feature and a logistic regression learner for each of the K classification tasks. This leads to K models predicting the likelihood that an unseen song belongs to class k. For example, if D_k represents all Jazz songs in the ground truth data set D, we learn a model separating these Jazz songs from songs of other genres, i.e. from $D \setminus D_k$. Using this model we are able to predict for a new song how "jazzy" it sounds, even if it is not a song from the Jazz genre itself. Note, that the method is by no means restricted to genre classes, any ground truth related to the sound properties can be used.

Using these likelihood predictions as new feature set reduces the amount of features from 40,000 to K. In our experiments we used genre classification data sets as the ground truth with $K < 10$. The predictions of the logistic regression models thus strongly compress the most relevant temporal statistics derived from the long song segments.

Figure 1 shows the overview of our proposed process. In the training phase a large number of short-term and long-term features is generated from the audio data. The regression models are trained for each musical aspect resulting in semantical features that can be used e.g. to train a classifier. For new audio data, only those short-term and long-term features need to be generated that have been found relevant by at least one regression learner. The music can be classified with the previously trained classifier, or a new classifier can be trained using the semantical features of the original training data. Alternatively, the features could be used for other music mining tasks like visualization of music collections or playlist generation.

5. EVALUATION

[6] `http://www.stat.rutgers.edu/~madigan/BBR`

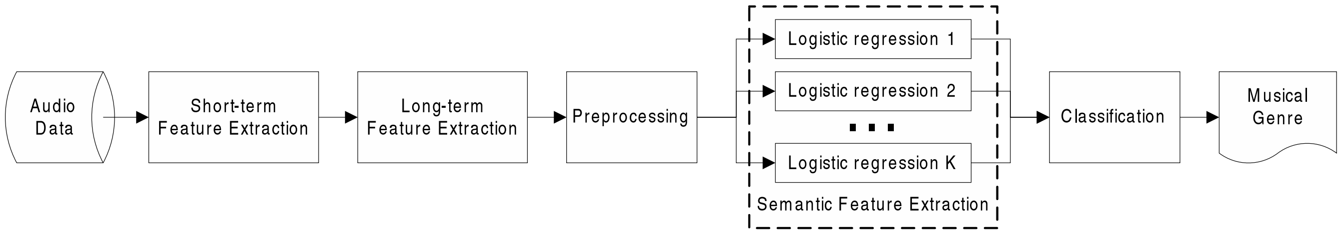


Figure 1: Proposed semantic modeling of music for music mining tasks like genre classification.

Table 4: Precision, recall, and number of selected features for the logistic regression models of each genre in the RADIO ground truth.

Genre	Precision	Recall	Features
Country	94.34	97.09	103
Dance	94.44	83.33	93
Jazz	100.00	97.03	146
Metal	98.91	88.35	121
Soul	89.80	84.62	168
Rap	83.02	87.13	139
World	89.77	80.00	133

In this section we present results on the real-world benchmark data sets described in section 3. First, we will discuss the learning of models and the influence of the features for different genre models. In a second part we select two of the data sets as ground truth and train specialized regression models in order to build new and comprehensible feature sets. We will evaluate the performance of the models learned from the semantic features and compare them to models learned from standard feature sets. Finally, we discuss the interpretability of the novel music descriptors.

5.1 Analysis of semantic audio features

The logistic regression learning of the genre ground truth worked very well within the RADIO and GTZAN data sets. Figure 2 shows the distribution of the output probabilities for the genre Metal in the RADIO data. For both the training and the disjunct test part of the data, the separation of Metal from the remaining music is clearly visible.

Table 4 summarizes the regression models for all seven genres of the RADIO data. The precision and recall values as measured on the test set are listed. The best performance was observed for the Jazz genre. The last columns show the number of long-term features picked out of the almost 40,000 candidate features. This can be interpreted as an indicator for the complexity of separating the genre from the remaining music. The model for Dance uses the fewest features, whereas Soul needs the most.

In order to generate the seven semantic features for this ground truth, the union of all selected long-term features would need to be extracted from new songs. There seem to exist many general purpose long-term features picked for several models, because the union of all features counts only 712 compared to the sum of 903. Table 5 lists the long-term features picked for 5 or 6 of the 7 models. The features are surprisingly simple, the temporal structure of the short-term features is only incorporated by differencing.

Table 5: Most frequently selected long-term features for the 7 models built with the RADIO ground truth.

Long-term feature	Selected
$kurt(\Delta(BandEnergyRatio))$	6×
$median(\Delta(SpectralRolloff))$	6×
$median(\Delta^2(SpectralRolloff))$	6×
$mean(\Delta^2(Mel_{28}))$	5×
$mean(\Delta^2(Mel_{33}))$	5×
$kurt(Mel_{34})$	5×

Table 6: Most influential long-term features per genre for RADIO ground truth.

Genre	Feature	Weight
Country	$mean(Chroma_F)$	0.48
Dance	$slope(ac(LowEnergy))$	-0.46
Jazz	$mean(\Delta(log(Chroma_{D\#})))$	0.41
Metal	$ac_1(Chroma_{D\#})$	-0.41
Soul	$mod_{1-2}(SpectralError)$	0.64
Rap	$std(abs(\Delta^2(Chroma_{G\#})))$	-0.68
World	$kurt_{5\%}(angles(PS_{2,1}(Mel_{20})))$	0.23

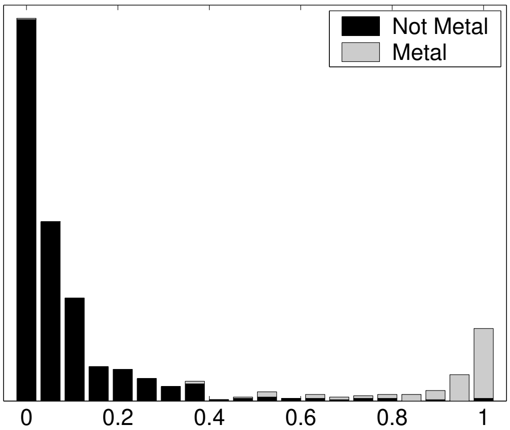
We further investigated which features had the largest absolute weights in the logistic regression models, indicating their relative importance in the decision for a genre (Table 6). Both very simple and quite complex features are among the most influential for the seven genres. For Country music the mean of the Chroma tone F has the largest positive weight, for Soul the modulation energy from 1-2Hz of the short-term feature SpectralError has a very large weight.

5.2 Genre classification

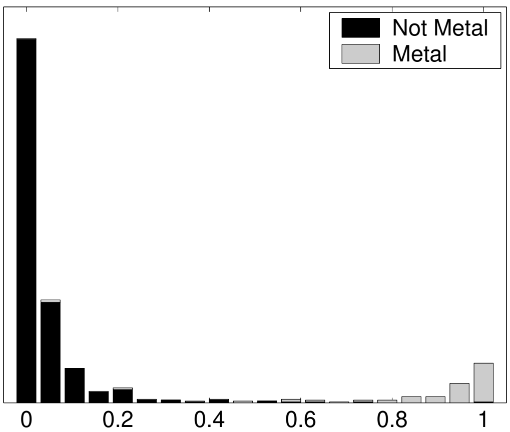
We compare the small and interpretable feature sets created from the logistic regression predictions with six previously published general purpose feature sets. We used the the 30 dimensional feature set of [38] extracted with the MARSYAS[38][7] software in Version 0.1 and the 72 dimensional feature set generated by Version 0.2. The features from [29] were extracted using the available toolbox [28][8]: *Spectrum Histogram* (SH, 1150 features), *Periodicity*

[7] `http://marsyas.sf.net`

[8] `http://www.oefai.at/~elias/ma`



(a) training set



(b) test set

Figure 2: Distribution of predictions from the logistic regression model trained with the Metal genre in the RADIO ground truth.

Histograms (PH, 2050 features), *Fluctuation Patterns* (FP, 1380 features). Finally, the 20 long-term features of the MusicMiner software were used. These features were selected from the same 40,000 candidate features according to the procedure described in [25].

Since we intend to measure the influence of the feature sets in contrast to the learning scheme abilities we use three learners with different learning properties for all feature set comparisons. These are a Support Vector Machine with linear kernel function (SVM) [31], a k-nearest neighbors learner with $k = 9$ (KNN) [12], and a decision tree learner (C4.5) [32]. All learning schemes are applied on the comparison feature sets extracted from the four datasets. We measure the classification accuracy for predicting the correct genre with help of a 10-fold cross validation. The results are presented in Figure 3. All classification experiments were performed with the freely available machine learning environment Yale[8][9].

Surprisingly, the combination of a linear support vector machine with the Marsyas-0.2 feature set outperforms all other combinations for all datasets. For KNN and C4.5 the Marsyas-0.2 and the MusicMiner features perform best.

[9] `http://yale.sf.net`

Since the training of the logistic regression models performed best for GTZAN and RADIO, we use these data sets as ground truth. We randomly divide the data sets in two parts with equal numbers of instances. We then use the logistic regression learner to create 10 and 7 specialized models respectively from one of the halves. These models are applied on both the other datasets and the half which was not used for training the regression models. Again, we use a 10-fold cross validation of SVM, KNN, and C4.5 to estimate the prediction accuracy by using these small feature sets of size 10 and 7. Figure 4 shows the results for both GTZAN and RADIO as ground truth data sets. The best results achieved with a SVM in combination with the Marsyas-0.2 features are also presented.

It can be seen that using our small and interpretable feature sets derived from the exhaustive set of temporal statistics features clearly outperforms the other feature sets at least on the test half of the same data set and is at least competitive for some of the other datasets. In most of the other cases the new features lead to results at least higher than the median of the results achieved by the comparison feature sets. Both facts are a clear indicator that the results achieved by our approach are at least comparable to the results achieved with traditional methods.

5.3 Interpretability

The k learned features can easily be interpreted since users usually have an idea of concepts like Jazz, Soul, or Rap. Figure 5 shows a decision tree for the genre classification data set MAB based on the ground truth of the RADIO data. This leads to rules like

`if a song does not sound like Rap in RADIO (`$\leq$` 0.34) but it sounds like Metal in RADIO (`$>$` 0.18) then it belongs to Rock in MAB`

or

`if a song does not sound like Rap and Metal in RADIO (`$\leq$` 0.34 and 0.18) but it sounds like Country, Jazz, and Soul in RADIO (`$>$` 0.03, 0.02 and 0.25) then the it belongs to Folk in MAB.`

Please note, that neither Rock nor Folk were part of the RADIO data set, they are explained in terms of their similarity to the songs of the clearly distinguishable genres of the RADIO data.

Figure 6 shows the decision tree for the test half of the radio data set. It can clearly be seen that in most cases the corresponding genre feature is used for classification, e.g.

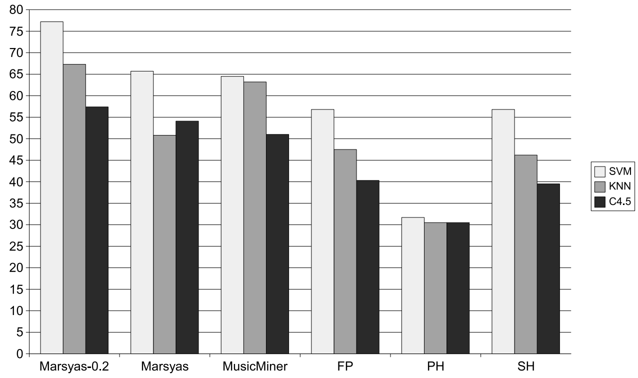
`if a song sounds like Country in RADIO (`$>$` 0.44) then it belongs to Country.`

However, in some cases not so intuitive decisions are generated. For example, the Jazz genre is explained by the Metal feature. We analyzed this and found that the information gain of the Metal feature set was slightly bigger than that of the Jazz feature causing the tree learner to seemingly pick the wrong descriptor.

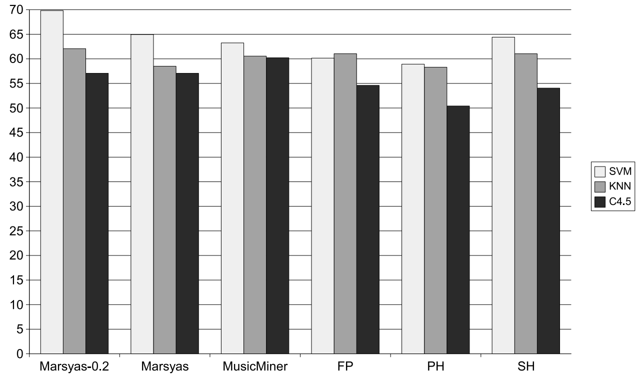
6. DISCUSSION

We presented a method for learning an arbitrary notion of music from a labeled set of training data. The resulting semantical features are better understandable than previously

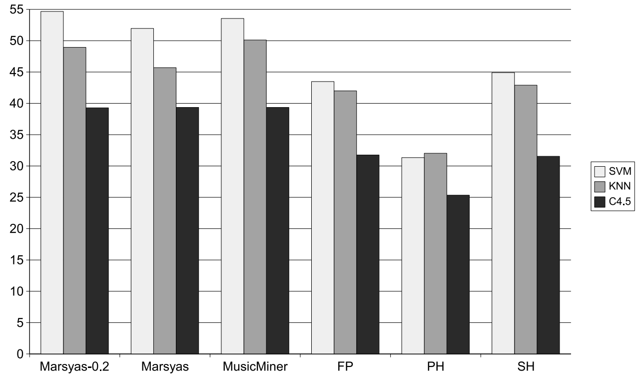
Figure 3: Accuracy for previously proposed feature sets. The used learning schemes were a Support Vector Machine with linear kernel (SVM), k-nearest neighbors (KNN) and a decision tree learner (C4.5). The results were evaluated on the data sets GTZAN (a), ISMIR04 (b), MAB (c), and RADIO (d).



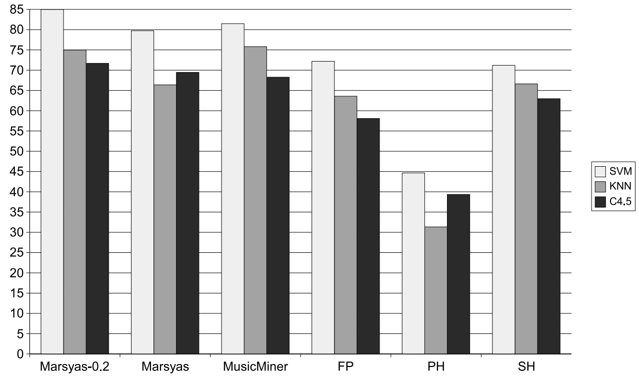
(a) GTZAN



(b) ISMIR04

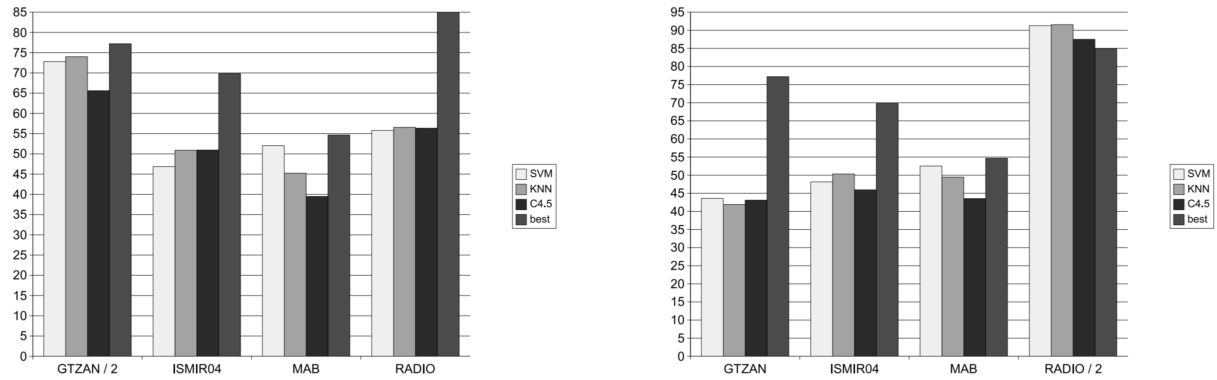


(c) MAB



(d) RADIO

Figure 4: Accuracy for different learning schemes using the feature generation approach discussed in this paper and the best accuracy achieved with traditional approachs (best of Figure 3). The ground truth data set were GTZAN (a) and RADIO (b).



(a) GTZAN ground truth

(b) RADIO ground truth

proposed features and were able to compete in the common genre classification problem. Other music mining tasks like recommendation or visualization could also profit from the higher understandability. The semantic features could be used to let the user control the emphasis put on certain musical aspects during the search. If the users provide a categorization of some music he knows well, our method could generate personalized features that describe *how much does this sound like other music that makes me happy*.

Interestingly, the genre ground truth of the RADIO data performed best within the collection and when applied to the other collections. We would like to emphasize that we did not put a lot of effort into creating this data, we simply relied on the consistency of several internet radio stations and only filtered out announcements.

We used genre ground truth for our evaluation, because it is most easily available in large quantities needed for the regression models. In principle, however, any ground truth related to the sound properties can be used, e.g., artist, album, timbre, mood, occasion, complexity, or intensity. If desired, users can define aspects that best describe their own musical preferences and provide training data in order to learn this subjective view of musical similarity. This further increases the interpretability of the models, since the features directly describe concepts the user is familiar with. Different features can be learned for multiple granularities, e.g. broadly acknowledged genres vs. sub-genres of Jazz that are only distinguishable by experts of the field. Recently, we have added a function to the MusicMiner software that allows the users to submit semantical ratings of musical aspects like mood to a web service. This way we hope to collect data for building models based on aspects other than genre.

Of course, other regression methods could just as well be used for learning the semantic features. One advantage of logistic regression is, that the numerical values do not need preprocessing for methods relying on distance calculations like k-nearest neighbor classification, k-Means clustering, or visualization with Emergent Self-Organizing Maps [42, 24].

The amount of candidate features is only limited by the computational resources. We believe, that by using more long-term features, the accuracy of our models can still be increased. More complex higher level features that are not formed by aggregating short-term features, like Beat Content [41], can also easily be added to the input of the regression models. The calculation of many long-term features can be quite time consuming, but the complete set only needs to be extracted for the training data. For the *RADIO* ground truth only 712 long-term features are need thereafter to determine the 7 semantic features. This enables real-time applications of music mining tasks in huge musical databases.

It would be interesting to investigate whether our approach of semantic feature generation can be applied in other areas where a large number of technical features is available, many of which might not be relevant. For example text mining (e.g. [6]) with large feature sets corresponding to words occurring in documents or video mining (e.g. [34]) where many features could be derived by combining short-term and long-term descriptions as we did for music.

7. SUMMARY

By plugging together many established data mining techniques we designed a system that provides understandable descriptions of music according to arbitrary notions of musical similarity. Exhaustive feature generation is used to capture many different aspects of the raw audio data that cannot be used directly. Feature selection and regression summarize the most relevant features for a particular aspect of music into a single number. This can be seen as a meta learning technique loosely related to stacking. The resulting low-dimensional vector based representations can efficiently be used for music mining tasks in like genre classification, recommendation, or visualization of music collections.

Acknowledgments: We thank Ingo Löhken, Michael Thies, Mario Nöcker, Christian Stamm, Niko Efthymiou, Martin Kümmerer, Timm Meyer, and Katharina Dobs for their help in the MusicMiner project. Fabian Mörchen was partly

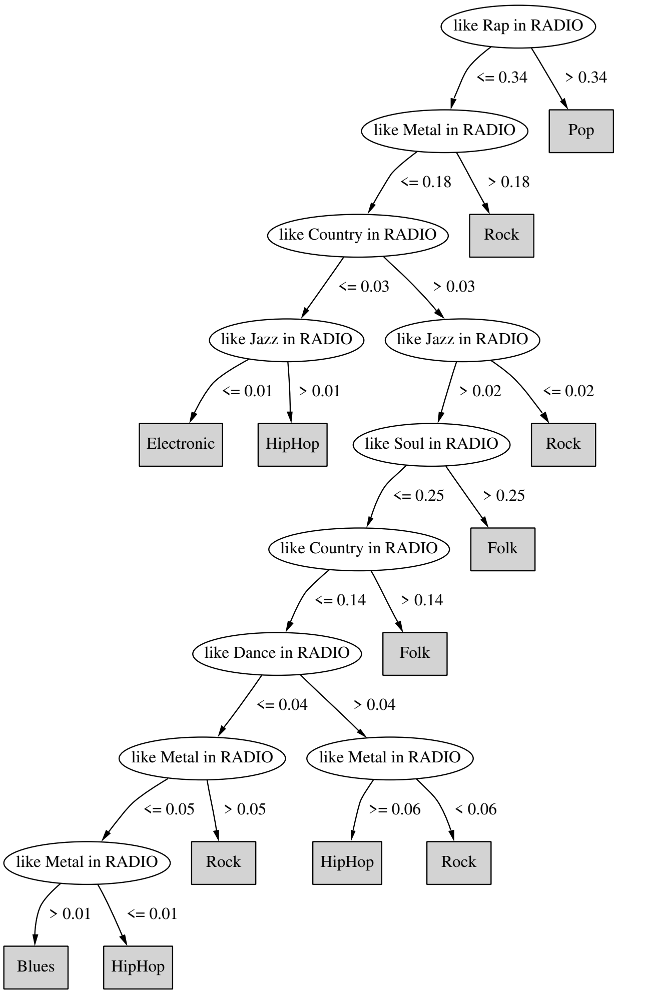


Figure 5: Learned decision tree from logistic regression predictions based on the RADIO data set for the data set MAB (see Section 5.2).

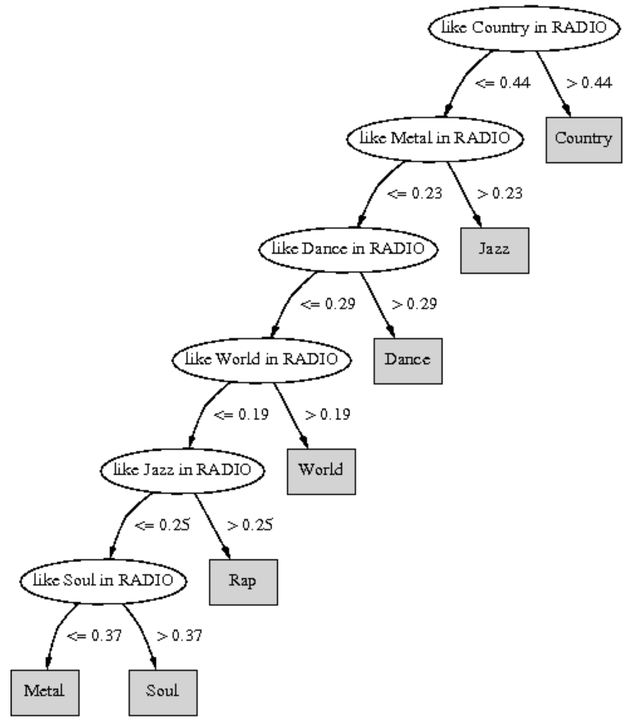


Figure 6: Learned decision tree from logistic regression predictions based on the training RADIO data set for the test data set RADIO (see Section 5.2).

supported by Siemens Corporate Research, Princeton, NJ, USA.

8. REFERENCES

[1] C. C. Aggarwal, A. Hinneburg, and D. A. Keim. On the surprising behavior of distance metrics in high dimensional space. In *Proc. Intl. Conf. on Database Theory*, pages 420–434, 2001.

[2] J.-J. Aucouturier and F. Pachet. Finding songs that sound the same. In *Proc. IEEE Benelux Workshop on Model based Processing and Coding of Audio*, pages 1–8, 2002.

[3] 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):1–13, 2004.

[4] J.-J. Aucouturier and F. Pachet. Tools and architecture for the evaluation of similarity measures: case study of timbre similarity. In *Proc. ISMIR*, 2004.

[5] A. Berenzweig, D. Ellis, and S. Lawrence. Anchor space for classification and similarity measurement of music. In *Proc. ICME*, pages I–29–32, 2003.

[6] M. W. Berry. *Survey of Text Mining : Clustering, Classification, and Retrieval*. Springer, 2003.

[7] T. G. Dietterich. Ensemble methods in machine learning. In J. Kittler and F. Roli, editors, *First International Workshop on Multiple Classifier Systems*, pages 1–15, 2000.

[8] S. Fischer, R. Klinkenberg, I. Mierswa, and O. Ritthoff. Yale: Yet Another Learning Environment – Tutorial. Technical Report CI-136/02, Collaborative Research Center 531, University of Dortmund, Germany, 2002.

[9] A. Genkin, D. D. Lewis, and D. Madigan. Large-scale bayesian logistic regression for text categorization. Technical report, DIMACS, 2004.

[10] M. Goto. A chorus-section detecting method for musical audio signals. In *Proc. IEEE ICASSP*, pages 437–440, 2003.

[11] G. Guo and S. Z. Li. Content-Based Audio Classification and Retrieval by Support Vector Machines. *IEEE Transaction on Neural Networks*, 14(1):209–215, 2003.

[12] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2001.

[13] P. Herrera, J. Bello, G. Widmer, M. Sandler, O. Celma, F. Vignoli, E. Pampalk, P. Cano, S. Pauws, and X. Serra. Simac: Semantic interaction with music audio contents. In *Proc. of the 2nd European Workshop on the Integration of Knowledge, Semantic and Digital Media Technologies*, 2005.

[14] H. Homburg, I. Mierswa, B. Moeller, K. Morik, and M. Wurst. A benchmark dataset for audio classification and clustering. In *Proc. ISMIR*, pages 528–531, 2005.

[15] H. Kantz and T. Schreiber. *Nonlinear Time Series Analysis*. Cambridge University Press, 1997.

[16] S. le Cessie and J. van Houwelingen. Ridge estimators in logistic regression. *Applied Statistics*, 41(1):191–201, 1992.

[17] D. Li, I. Sethi, N. Dimitrova, and T. McGee. Classification of general audio data for content-based retrieval. *Pattern Recognition Letters*, 22:533–544, 2001.

[18] T. Li, M. Ogihara, and Q. Li. A comparative study on content-based music genre classification. In *Proc. ACM SIGIR*, pages 282–289, 2003.

[19] B. Logan and A. Salomon. A music similarity function based on signal analysis. In *IEEE Intl. Conf. on Multimedia and Expo*, page 190, 2001.

[20] M. McKinney and J. Breebaart. Features for audio and music classification. In *Proc. ISMIR*, pages 151–158, 2003.

[21] A. Meng, P. Ahrendt, and J. Larsen. Improving music genre classification by short-time feature integration. In *Proc. IEEE ICASSP*, pages 497–500, 2005.

[22] I. Mierswa and K. Morik. Automatic feature extraction for classifying audio data. *Machine Learning Journal*, 58:127–149, 2005.

[23] B. Moore and B. Glasberg. A revision of zwickers loudness model. *ACTA Acustica*, 82:335–345, 1996.

[24] F. Mörchen, A. Ultsch, M. Nöcker, and C. Stamm. Databionic visualization of music collections according to perceptual distance. In *Proc. ISMIR*, pages 396–403, 2005.

[25] F. Mörchen, A. Ultsch, M. Thies, and I. Löhken. Modelling timbre distance with temporal statistics from polyphonic music. *IEEE TSAP*, 14(1), 2006.

[26] F. Mörchen, A. Ultsch, M. Thies, I. Löhken, M. Nöcker, C. Stamm, N. Efthymiou, and M. Kümmerer. MusicMiner: Visualizing timbre distances of music as topograpical maps. Technical report, CS Dept., Philipps-University Marburg, Germany, 2005.

[27] F. Pachet and A. Zils. Evolving automatically high-level music descriptors from acoustic signals. In *Proc. Intl. Symposium on Computer Music Modeling and Retrieval*, 2003.

[28] E. Pampalk. A Matlab toolbox to compute music similarity from audio. In *Proc. ISMIR*, 2004.

[29] E. Pampalk, S. Dixon, and G. Widmer. On the evaluation of perceptual similarity measures for music. In *Proc. Intl. Conf. on Digital Audio Effects*, pages 6–12, 2003.

[30] E. Pampalk, A. Rauber, and D. Merkl. Content-based organization and visualization of music archives. In *Proc. ACM Multimedia*, pages 570–579, 2002.

[31] J. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, chapter 12. MIT-Press, 1999.

[32] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Machine Learning. Morgan Kaufmann, San Mateo, CA, 1993.

[33] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, 1993.

[34] C. Snoek and M. Worring. Multimodal video indexing: A review of the state-of-the-art. *Multimedia Tools and Applications*, 25(1):5–35, 2005.

[35] R. Stenzel and T. Kamps. Improving content-based similarity measures by training a collaborative model. In *Proc. ISMIR 2005*, pages 264–271, 2005.

[36] F. Takens. Dynamical systems and turbulencs. In D. Rand and L. Young, editors, *Lecture Notes in Mathematics*, volume 898, pages 366–381. Springer, 1981.

[37] R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Royal Statistical Soc. B.*, 58:267–288, 1996.

[38] G. Tzanetakis and P. Cook. Marsyas: A framework for audio analysis. *Organised Sound*, 4(30):169–175, 2000.

[39] G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE TSAP*, 10(5):293–302, 2002.

[40] G. Tzanetakis, G. Essl, and P. Cook. Automatic musical genre classification of audio signals. In *Proc. ISMIR*, pages 205–210, 2001.

[41] G. Tzanetakis, G. Essl, and P. Cook. Human perception and computer extraction of beat strength. In *Proc. Intl. Conf. on Digital Audio Effects*, 2002.

[42] A. Ultsch. Self-organizing neural networks for visualization and classification. In *Proc. Conf. German Classification Society*, 1992.

[43] K. West and S. Cox. Features and classifiers for the automatic classification of musical audio signals. In *Proc. ISMIR*, 2004.

[44] D. H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.

[45] C. Xu, N. Maddage, and X. Shao. Musical genre classification using support vector machines. In *Proc. IEEE ICASSP*, pages 429–432, 2003.

[46] T. Zhang and C. Kuo. Content-based Classification and Retrieval of Audio. In *Conf. on Advanced Signal Processing Algorithms, Architectures, and Implementations VIII*, 1998.

[47] E. Zwicker and S. Stevens. Critical bandwidths in loudness summation. *The Journal of the Acoustical Society of America*, 29(5):548–557, 1957.