

**Book *Advances in Music Information Retrieval***


Chapter *Multimodal Aspects of Music Retrieval: Audio, Song Lyrics – and Beyond?*

This is a self-archived pre-print version of this article.

The final publication is available at Springer via  
[https://doi.org/10.1007/978-3-642-11674-2\\_15](https://doi.org/10.1007/978-3-642-11674-2_15).




# Multimodal Aspects of Music Retrieval: Audio, Song Lyrics – and beyond?

Rudolf Mayer  and Andreas Rauber

**Abstract** Music retrieval is predominantly seen as a problem to be tackled in the acoustic domain. With the exception of symbolic music retrieval and score-based systems, which form rather separate sub-disciplines on their own, most approaches to retrieve recordings of music by content rely on different features extracted from the audio signal. Music is subsequently retrieved by similarity matching, or classified into genre, instrumentation, artist or other categories. Yet, music is an inherently multimodal type of data. Apart from purely instrumental pieces, the lyrics associated with the music are as essential to the reception and the message of a song as is the audio. Album covers are carefully designed by artists to convey a message that is consistent with the message sent by the music on the album as well as by the image of a band in general. Music videos, fan sites and other sources of information add to that in a usually coherent manner. This paper takes a look at recent developments in multimodal analysis of music. It discusses different types of information sources available, stressing the multimodal character of music. It then reviews some features that may be extracted from those sources, focussing particularly on audio and lyrics as sources of information. Experimental results on different collections and categorisation tasks will round off the chapter. It shows the merits and open issues to be addressed to fully benefit from the rich and complex information space that music creates.

---

Rudolf Mayer   
Institute of Software Technology and Interactive Systems, Vienna University of Technology  
e-mail: mayer@ifs.tuwien.ac.at

Andreas Rauber  
Institute of Software Technology and Interactive Systems, Vienna University of Technology  
e-mail: rauber@ifs.tuwien.ac.at

## 1 Introduction

Multimedia data by definition incorporates multiple types of content. However, often a strong focus is put on one view only, disregarding many other opportunities and exploitable modalities. In the same way as video, for instance, incorporates visual, auditory, and text info (in the case of subtitles or extra information about the current programme via TV text and other channels), music data itself is not limited solely to its sound. Yet, a strong focus is put on audio based feature sets throughout the music information retrieval community, as music perception itself is based on sonic characteristics to a large extent. For many people, acoustic content is the main property of a song and makes it possible to differentiate between acoustic styles. For many examples or even genres this is true, for instance ‘Hip-Hop’ or ‘Techno’ music being dominated by a strong bass. Specific instruments very often define different types of music – once a track contains trumpet sounds it will most likely be assigned to genres like ‘Jazz’, traditional Austrian/German ‘Blasmusik’, ‘Classical’, or ‘Christmas’.

However, a great deal of information is to be found in extra information in the form of text documents, be it about artists, albums, or song lyrics. Many musical genres are rather defined by the topics they deal with than a typical sound. ‘Christmas’ songs, for instance, are spread over a whole range of musical genres. Many traditional ‘Christmas’ songs were interpreted by modern artists and are heavily influenced by their style; ‘Punk Rock’ variations are recorded as well as ‘Hip-Hop’ or ‘Rap’ versions. What all of these share, though, is a common set of topics to be sung about. Another example is ‘Christian Rock’, which has a sound indistinguishable from other Rock music, but has highly religious topics (the same holds true for ‘Christian Hip-Hop’). These simple examples show that there is a whole level of semantics inherent in song lyrics, that can not be detected by audio based techniques alone.

We assume that a song’s text content can help in better understanding its meaning. In addition to the mere textual content, song lyrics exhibit a certain structure, as they are organised in blocks of choruses and verses. Many songs are organised in rhymes, patterns which are reflected in a song’s lyrics and easier to detect from text than audio. Whether or not rhyming structures occur at all, and the level of complexity of the patterns used, may be highly characteristic for certain genres. In some cases, for example when thinking about very ‘ear-catching’ songs, maybe even the simplicity of rhyme structures are the common denominator.

For similar reasons, musical similarity can also be defined on textual analysis of certain parts-of-speech (POS) characteristics. Quiet or slow songs could, for instance, be discovered by rather descriptive language which is dominated by nouns and adjectives, whereas we assume a high number of verbs to express the nature of lively songs. In this paper, we further show the influence of so called text statistic features on song similarity. We employ a range of simple statistics such as the average word or line lengths as descriptors. Analogously to the common beats-per-minute (BPM) descriptor in audio analysis, we introduce the words-per-minute (WPM) measure to identify similar songs. The rationale behind WPM is that it can

capture the ‘density’ of a song and its rhythmic sound in terms of similarity in audio and lyrics characteristics.

We therefore stress the importance of taking into account several of the aforementioned properties of music by means of a combinational approach. We want to point out that there is much to be gained from such a combinational approach as single genres may be best described in different feature sets. Musical genre classification therefore is heavily influenced by these modalities and can yield better overall results. We show the applicability of our approach with a detailed analysis of both the distribution of text and audio features, and genre classification on two test collections. One of our test collections consists of manually selected and cleansed songs subsampled from a real-world collection. We further use a larger collection which again is subsampled, but not manually cleansed, to show the stability of our approach.

This remainder of this paper is structured as follows. We start by giving an overview on related work in Section 2. We then give a detailed description of our approach and the feature sets we use for analysing song lyrics and audio tracks alike in Section 3. In Section 4 we apply our techniques to several audio corpora. We provide a summary of previous as well as novel results for the musical genre classification task, and a wide range of experimental settings. Finally, we analyse our results, conclude, and give a short outlook on future research in Section 5.

## 2 Related Work

Music information retrieval is a discipline of information retrieval, concerned with adequately accessing (digital) audio. Its major research topics include, but are not limited to, musical genre classification (and classification into other types of categories, such as mood or situations), similarity retrieval, or music analysis and knowledge representation. Comprehensive overviews of music information retrieval research are given in [8, 27].

The still dominant method of processing audio files in music information retrieval is by analysis of the audio signal, which is computed from plain wave files or via a preceding decoding step from other wide-spread audio formats such as MP3 or the (lossless) FLAC format. A wealth of different descriptive features for the abstract representation of audio content have been presented. Early overviews on content-based music information retrieval and experiments are given in [10] and [37, 39], focussing mainly on automatic genre classification of music.

Mel-Frequency Cepstral Coefficients (MFCC) [32] are a perceptually motivated set of features developed in context of speech recognition. The Mel scale, which is a perceptual scale found empirically through human listening tests, and models perceived pitch distances, is applied to the logarithmic spectrum before applying a discrete cosine transform (or an inverse Fourier transform) to obtain the MFCCs. An investigation about their adoption in the MIR domain was presented in [19]. Content-based audio retrieval based on K-Means clustering of MFCC features is

performed in [21]. A comparison of MFCC and MPEG-7 features on sports audio classification is presented in [40].

Daubechies Wavelet Coefficient Histograms as a feature set suitable for music genre classification are proposed in [16]. The feature set characterises amplitude variations in the audio signal.

Chroma features [11] extract the harmonic content (e.g. keys, chords) of music by computing the spectral energy present at frequencies that correspond to each of the 12 notes in a standard chromatic scale.

The MARSYAS system [37], besides new graphical user interfaces for browsing and interacting with audio signals, introduces a number of new algorithms for audio description: a general multifeature audio texture segmentation methodology, feature extraction from MP3 compressed data, beat detection based on the discrete Wavelet transform and musical genre classification combining timbral, rhythmic and harmonic features.

The Moving Picture Experts Group (MPEG) released the MPEG-7 standard, which defines the Multimedia Content Description Interface, and is a standard for description and search of audio and visual content. Part 4 of said standard describes 17 low-level audio temporal and spectral descriptors, divided into seven classes, including silence. Some of the features are based on basic wave-form or spectral information, while others use harmonic or timbral information. In [1] these features are used for audio fingerprinting, i.e. using signatures based on various properties of audio signal for the robust identification of audio material. A classification approach with MPEG-7 features is done in [6].

Rhythm Patterns [34, 29] are a set of audio features which model modulation amplitudes on critical frequency bands. To this end, they consider and employ a set of psycho-acoustic models. Two other feature sets have been derived from and are based on different parts of the computation of the Rhythm patterns, namely the Rhythm Histograms and Statistical Spectrum Descriptors [17] feature sets.

In this paper, the MFCC, Marsyas, Chroma, Rhythm Patterns, Rhythm Histograms and Statistics Spectrum Descriptors are combined with and compared to our set of lyrics features. Therefore, these audio feature sets will be described in more detail in Section 3.1.

Several research teams have further begun working on adding textual information to the retrieval process, predominantly in the form of song lyrics and an abstract vector representation of the term information contained in text documents. A semantic and structural analysis of song lyrics is conducted in [22]. It focuses on aspects such as structure detection, e.g. verses and chorus, classification into thematic categories such as ‘love’, ‘violent’, ‘christian’, and similarity search. The correlation between artist similarity and song lyrics is studied in [20]. It is pointed out that acoustic similarity is superior to textual similarity, yet a combination of both approaches might lead to better results. A promising approach targeted at large-scale recommendation engines is presented in [14]. Lyrics are gathered from multiple sources on the Web, and are subsequently aligned to each other for matching sequences, to filter out er-

rors like typing errors, or retrieved parts not actually belonging to the lyrics of the song, such as commercials.

Also, the analysis of karaoke music is an interesting new research area. A multi-modal lyrics extraction technique for tracking and extracting karaoke text from video frames is presented in [42]. Some effort has also been spent on the automatic synchronisation of lyrics and audio tracks at a syllabic level [12]. A multi-modal approach to query music, text, and images with a special focus on album covers is presented in [4]. Other cultural data is included in the retrieval process e.g. in the form of textual artist or album reviews [3]. Cultural data is also used to provide a hierarchical organisation of music collections on the artist level in [28]. The system describes artists by terms gathered from web search engine results.

Another area where lyrics have also been employed is the field of emotion detection and classification, for example [41], which aims at disambiguating music emotion with lyrics and social context features. More recent work combined both audio and lyrics-based feature for mood classification [15].

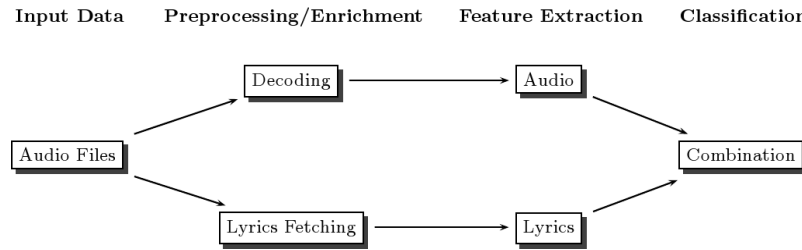
In [13], additional information like web data and album covers are used for labelling, showing the feasibility of exploiting a range of modalities in music information retrieval. A three-dimensional musical landscape via a Self-Organising Maps (SOMs) is created and applied to small private music collections. Users can then navigate through the map by using a video game pad. An application of visualisation techniques for lyrics and audio content based on employing two separate SOMs is given in [26]. It demonstrates the potential of lyrics analysis for clustering collections of digital audio. The similarity of songs is visualised according to both modalities, and a quality measure with respect to the differences in distributions across the two maps is computed, in order to identify interesting genres and artists.

Experiments on the concatenation of audio and bag-of-words features were reported in [25]. The results showed potential for dimensionality reduction when using different types of features.

First results for genre classification using the rhyme and style features used later in this paper are reported in [24]; these results particularly showed that simple lyrics features may well be worthwhile. This approach has further been extended on two bigger test collections, and to combining and comparing the lyrics features with audio features in [23].

### 3 Employed Feature Sets

Figure 1 shows an overview of the processing architecture. We start from plain audio files. The preprocessing/enrichment step involves decoding of audio files to plain wave format as well as lyrics fetching. We then apply the audio and lyrics-based feature extraction described in the following subsections. Finally, the results of both feature extraction processes are used for musical genre classification.



**Fig. 1** Processing architecture for combined audio and lyrics analysis stretching from a set of plain audio files to combined genre classification

### 3.1 Audio Features

In our study, we employ several different sets of features extracted from the audio content of the songs, to compare them to and combine them with our newly designed set of features based on the song lyrics. To give comprehensive evidence that our feature set can improve classification results of audio-only feature sets, we extended the experiments presented in [23], which made use of the Rhythm Patterns, Statistical Spectrum Descriptors, and Rhythm Histograms audio feature sets. To those, we add analysis of the combination of the lyrics-based features with other popular and widely used feature sets, namely the Mel Frequency Cepstral Coefficients (MFCCs), MARSYAS and Chroma features. All these feature sets will be described below.

#### 3.1.1 MFCC Features

Mel Frequency Cepstral Coefficients (MFCCs) originated in research for speech processing [32], and soon gained popularity in the field of music information retrieval [19]. A cepstrum is defined as the Discrete Cosine Transform (DCT) or inverse Fourier transform of the logarithm of the spectrum. If the Mel scale is applied to the logarithmic spectrum before applying the DCT (or inverse Fourier transform), the result is called Mel Frequency Cepstral Coefficients. The Mel scale is a perceptual scale that models perceived pitch distances, and was found empirically through human listening tests. With increasing frequency, the intervals in Hz producing equal increments in perceived pitch are getting larger and larger. Thus, the Mel scale is approximately a logarithmic scale; it corresponds more closely to the human auditory system than the linearly spaced frequency bands of a spectrum. A related scale is e.g. the Bark Scale, used in the Rhythm Patterns features (c.f. Section 3.1.4). From the MFCCs, commonly only the first few (for instance 5 to 20) Coefficients are used as features. In this work, we use the MFCC features extracted by the MARSYAS system, which provides four statistical values (means and



variances over a texture window of one second) for the first 13 coefficients, thus resulting in 52 dimensions.

### 3.1.2 MARSYAS Features

The MARSYAS system [37] is a software framework for audio analysis and provides a number of feature extractors, all of which compute statistics over a texture window of approximately one second.

The *Short-Time Fourier Transform (STFT) Spectrum based Features* provide standard temporal and spectral low-level features, such as Spectral Centroid, Spectral Rolloff, Spectral Flux, Root Mean Square (RMS) energy and Zero Crossings.

A set of *MPEG Compression based features* is extracted directly from MPEG compressed audio data (e.g. from mp3 files) [38]. This approach utilises the fact that MPEG compression already performs a lot of analysis in the encoding stage, including a time-frequency analysis. The spectrum is divided into 32 sub-bands of equal size, via an analysis filterbank, wherefrom features such as the centroid, rolloff, spectral flux and RMS are directly computed from. Note that these features are not equal to the MPEG-7 standard features.

The *Wavelet Transform* is an alternative to the Fourier Transform, overcoming the trade-off between time and frequency resolution. It provides low frequency resolution and high time resolution for high frequency ranges, while in low frequency ranges, it provides high frequency and lower time resolution. This is a closer representation of the human perception of a sound. A set of features is extracted by computing the mean absolute values and standard deviation of the coefficients in each frequency band, and ratios of the mean absolute values between adjacent bands. The features represent ‘sound texture’ and provide information about the frequency distribution of the signal and its evolution over time.

For the *Beat Histogram* computation, a Discrete Wavelet Transform, which decomposes the signal into octave frequency bands, is applied before a time-domain amplitude envelope extraction and periodicity detection. The time domain amplitude envelope are extracted separately for each band. The sum of the normalised envelopes is then processed through an autocorrelation function to detect the dominant periodicities of the signal. The amplitude values of the dominant peaks are then accumulated over the whole song into the Beat Histogram, which not only captures the dominant beat in a sound, but more detailed information about the rhythmic content of a piece of music. The relative amplitude (of the sum of amplitudes) of the first and second peak, the ratio of the amplitude of the second to the first peak, the period of the first and second beat (in beats per minute), and the overall sum of the histogram, as indication of beat strength, are computed as features.

The *Pitch Histogram* feature computation decomposes the signal into two frequency bands (below and above 1000 Hz). For each band, amplitude envelopes are extracted, which are then summed up and an autocorrelation function is used to detect the main pitches. The three dominant peaks are accumulated into a histogram, where each bin corresponds to a musical note. The histogram thus contains infor-

mation about the pitch range of a piece of music. A folded version of the histogram, obtained by mapping the notes of all octaves onto a single octave, contains information about the pitch classes or the harmonic content. The amplitude of the maximum peak of the folded histogram (i.e. magnitude of the most dominant pitch class), the period of the maximum peak of the unfolded (i.e. octave range of the dominant pitch) and folded histogram (i.e. main pitch class), the pitch interval between the two most prominent peaks of the folded histogram (i.e. main tonal interval relation) and the overall sum of the histogram are computed as features.

### 3.1.3 Chroma Features

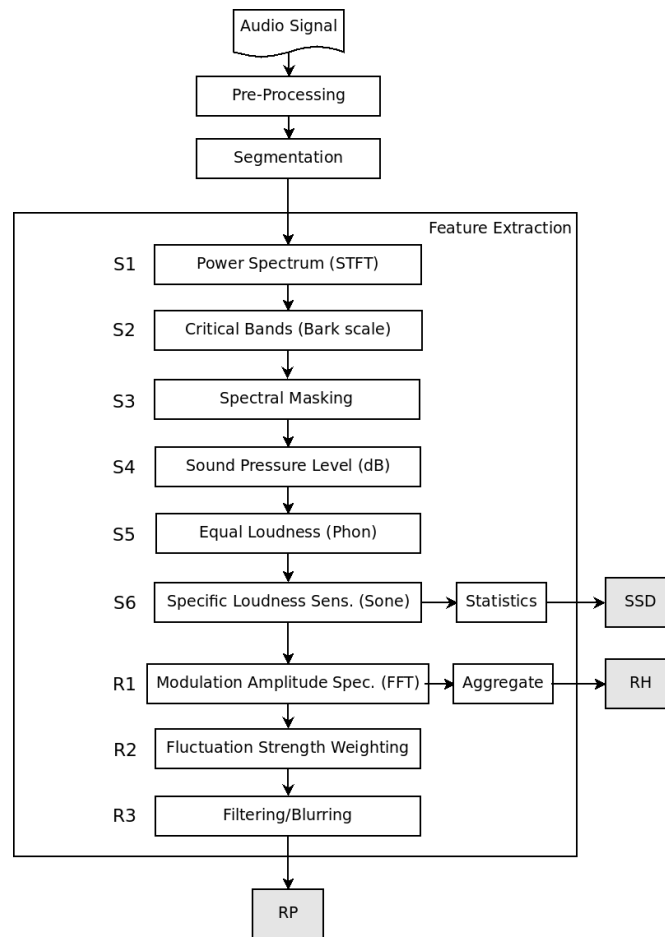
Chroma features aim at representing the harmonic content (e.g. keys, chords) of a short-time window of audio. The Chroma vector is a perceptually motivated feature vector[11]. It uses the concept of *chroma* in the cyclic helix representation of musical pitch perception [36]. Chroma therein refers to the position of a pitch within an octave. The chroma vector thus represents magnitudes in twelve pitch classes in a standard chromatic scale (e.g., black and white keys within one octave on a piano). The feature vector is extracted from the magnitude spectrum by using a short-time Fourier transform (STFT). We specifically employ the feature extractor implemented in the MARSYAS system, which computes four statistical values (means and variances over a texture window of one second), for each of the 12 chromatic notes, thus finally resulting in a 48-dimensional feature vector.

### 3.1.4 Rhythm Patterns

Rhythm Patterns (RP), also called Fluctuation patterns, are a feature set for handling audio data based on analysis of the spectral audio data and psycho-acoustic transformations [33, 17]. The feature set has been employed e.g. in the SOM-enhanced jukebox (SOMeJB) [30] digital music library system. Rhythm patterns are basically a matrix representation of fluctuations on several critical bands. An overview of the computational steps is given in Figure 2, which also depicts the process for obtaining the Statistical Spectrum Descriptions and Rhythm Histograms, which are derived from the Rhythm Patterns features, and skip or modify some of the processing steps; further, they exhibit a different feature dimensionality, and represent different aspects of the audio signal.

If needed, a set of preprocessing steps is applied before the actual feature computation: multiple channels are averaged to one, and the audio is segmented into parts of six seconds. Often, it can be of advantage to leave out possible lead-in and fade-out segments, which might greatly differ from the rest of the song. Depending on the processing capability available, also further segments maybe be skipped, e.g. only processing every third segment.

The feature extraction process for a Rhythm Pattern is then composed of two stages, indicated as steps *S1–S6* and *R1–R3* in Figure 2. First, the spectrogram of the



**Fig. 2** Steps of the feature extraction process for Rhythm Patterns (RP), Statistical Spectrum Descriptors (SSD), and Rhythm Histograms (RH)

audio is computed for each segment, utilising the short time Fast Fourier Transform (STFT), and applying a Hanning window (cf. S1). Next we employ the Bark scale, a perceptual scale that groups frequencies to *critical bands* according to perceptual pitch regions. Applying the scale to the spectrograms results in an aggregation to 24 frequency bands (S2). A Spectral Masking spreading function is applied to the signal, which models the occlusion of one sound by another sound (S3). Then, the Bark scale spectrogram is transformed into the decibel scale (S4), and further psycho-acoustic transformations are applied: computation of the Phon scale (S5) incorporates equal loudness curves, which account for the different perception of loudness at different frequencies. Subsequently, the values are transformed into the unit Sone (S6), which relates to the Phon scale in the way that a doubling on the Sone scale sounds to the human ear like a doubling of the loudness. This results in a

psycho-acoustically modified Sonogram representation that reflects human loudness sensation.

In the second stage, the varying energy on a critical band in the Bark-scale Sonogram is regarded as a modulation of the amplitude over time. A discrete Fourier transform is applied to this Sonogram, resulting in a time-invariant spectrum of loudness amplitude modulation per modulation frequency for each individual critical band (R1). After additional weighting (R2) and smoothing steps using a gradient filter and Gaussian smoothing (R3), a Rhythm Pattern finally exhibits the magnitude of modulation for 60 frequencies on 24 bands, and has thus 1440 dimensions.

In order to summarise the characteristics of an entire piece of music, the median of the Rhythm Patterns of the six-second segments is computed.

### 3.1.5 Statistical Spectrum Descriptors

Statistical Spectrum Descriptors (SSD) features are derived based on the first stage of the Rhythm Patterns computation, i.e. on the Bark-scale representation of the frequency spectrum (cf. steps S1–S6 in Figure 2). In order to describe fluctuations within the critical bands, from this representation of perceived loudness, seven statistical measures are subsequently computed for each segment per critical band: the mean, median, variance, skewness, kurtosis, min- and max-values, resulting in a Statistical Spectrum Descriptor for a segment. The SSD feature vector for a piece of audio is then again calculated as the median of the descriptors of its segments.

In contrast to the Rhythm Patterns feature set, the dimensionality of the feature space is much lower: SSDs have  $24 \times 7 = 168$  instead of 1440 dimensions, and this at matching performance regarding genre classification accuracies [17], on specific data sets even outperforming the Rhythm Patterns [18].

### 3.1.6 Rhythm Histogram Features

The Rhythm Histogram (RH) features are capturing rhythmical characteristics in a piece of music. Contrary to the Rhythm Patterns and the Statistical Spectrum Descriptor, information is not stored per critical band. Instead, early in the second stage of the RP calculation process (after step R1 in Figure 2), the magnitudes of each modulation frequency of all 24 critical bands are summed up, forming a histogram of 60 bins of ‘rhythmic energy’ per modulation frequency between 0.168 and 10 Hz. For a given piece of music, the Rhythm Histogram feature set is again calculated by taking the median of the histograms of every single segment processed. Rhythm Histogram features represent similar information as the Beat Histogram of MARSYAS, but have a different extraction approach.

We further utilise the beats per minute (BPM) feature, computed from the modulation frequency of the peak of a Rhythm Histogram, to give a comparison to the lyrics-based words per minute (WPM) feature (cf. Section 3.2.2).

## 3.2 Lyrics Features

In this section we describe the four types of lyrics features we use in the experiments throughout the remainder of the paper: a) bag-of-words features computed from tokens (terms) occurring in documents, b) rhyme features taking into account the rhyming structure of lyrics, c) features considering the distribution of certain parts-of-speech, and d) text statistics features covering average numbers of words and particular characters. The latter three feature sets are referred to as rhyme and style features.

### 3.2.1 Bag-Of-Words

Classical bag-of-words indexing at first tokenises all text documents in a collection, most commonly resulting in a set of words representing each document. Let the number of documents in a collection be denoted by  $N$ , each single document by  $d$ , and a term or token by  $t$ . Accordingly, the *term frequency*  $tf(t, d)$  is the number of occurrences of term  $t$  in document  $d$  and the *document frequency*  $df(t)$  the number of documents term  $t$  appears in. From this, an *inverse document frequency*  $idf$  can be computed.

The process of assigning weights to terms according to their importance or significance for the classification is called ‘term-weighing’. The basic assumptions are that terms which occur very often in a document are more important for classification, whereas terms that occur in a high fraction of all documents are less important. The weighing we rely on is the most common model of *term frequency times inverse document frequency* [35], computed as:

$$tf \times idf(t, d) = tf(t, d) \cdot \ln(N/df(t)) \quad (1)$$

This results in vectors of weight values for each document  $d$  in the collection, i.e. each song lyrics document. This representation also introduces a concept of similarity, as lyrics that contain a similar vocabulary are likely to be semantically related. We do not perform term stemming in this setup, as earlier experiments showed only negligible differences for stemmed and non-stemmed features [24]; the rationale behind using non-stemmed terms is the occurrence of slang language in some genres, which we aim to preserve.

Selecting all terms present in a document collection will in most cases yield a vocabulary too large to be adequately processed by machine learning algorithms. Further, some terms might rather add noise than helping to distinguish documents from different genres. Thus, *feature (or term) selection* is an important pre-processing step. In this work, we employ a *frequency thresholding* technique: we omit terms that occur too frequent, and thus are likely stop-words, and terms that occur in too few documents, and therefore likely have less discriminative power.

**Table 1** Overview of text statistic features

Feature Name	Description
exclamation mark, colon, quote, comma, question mark, dot, hyphen, semicolon	simple counts of occurrences
d0 - d9	occurrences of digits
WordsPerLine	words / number of lines
UniqueWordsPerLine	unique words / number of lines
UniqueWordsRatio	unique words / words
CharsPerWord	number of chars / number of words
WordsPerMinute	the number of words / length of the song

### 3.2.2 Text Statistic Features

Text documents can also be described by simple statistical measures based on term (word) or character frequencies. Measures such as the average length of words or the ratio of unique words in the vocabulary capture aspects of the complexity of the texts, and are expected to vary over different genres. Further, the usage of punctuation marks such as exclamation or question marks may be specific for some genres. We further expect some genres to make increased use of apostrophes when omitting the correct spelling of word endings. The list of extracted features is given in Table 1.

All features that simply count character occurrences are normalised by the number of words of the song text to accommodate for different lyrics lengths. ‘WordsPerLine’ and ‘UniqueWordsPerLine’ describe the words per line and the unique number of words per line. The ‘UniqueWordsRatio’ is the ratio of the number of unique words and the total number of words. ‘CharsPerWord’ denotes the simple average number of characters per word. The last feature, ‘WordsPerMinute’ (WPM), is computed analogously to the well-known beats-per-minute (BPM) value<sup>1</sup>. Even though the computation is similar, the two features may still take very different values in various genres – as such, both ‘Hip-Hop’ and e.g. ‘Techno’ music may have similar BPM, but the latter generally way less song text, and thus much lower WPM values.

### 3.2.3 Part-of-Speech Features

Part-of-speech tagging is a lexical categorisation or grammatical tagging of words according to their definition and the textual context they appear in. Different part-of-speech categories are for example nouns, verbs, articles or adjectives. We presume that different genres will differ also in the category of words they are using, and therefore we additionally extract several part of speech descriptors from the lyrics. To this end, we employ the ‘LingPipe’ suite of libraries<sup>2</sup>. We in particular count the

<sup>1</sup> Actually we use the ratio of the number of words and the song length in seconds to keep feature values in the same range. Hence, the correct name would be ‘WordsPerSecond’, or WPS.

<sup>2</sup> <http://alias-i.com/lingpipe/>

**Table 2** Rhyme features for lyrics analysis

Feature Name	Description
Rhymes-AA	A sequence of two (or more) rhyming lines ('Couplet')
Rhymes-AABB	A block of two rhyming sequences of two lines ('Clerihew')
Rhymes-ABAB	A block of alternating rhymes
Rhymes-ABBA	A sequence of rhymes with a nested sequence ('Enclosing rhyme')
RhymePercent	The percentage of blocks that rhyme
UniqueRhymeWords	The fraction of unique terms used to build the rhymes

numbers of: *nouns, verbs, pronouns, relational pronouns* (such as 'that' or 'which'), *prepositions, adverbs, articles, modals, and adjectives*. To account for different document lengths, all of these values are normalised by the number of words of the respective lyrics document.

### 3.2.4 Rhyme Features

Rhyme denotes the the consonance or similar sound of two or more syllables or whole words. This linguistic style is most commonly used in poetry and songs. The rationale behind the development of rhyme features is that different genres of music should exhibit different styles of lyrics. We assume the rhyming characteristics of a song to be given by the degree and form of the rhymes used. 'Hip-Hop' or 'Rap' music, for instance, makes heavy use of rhymes, which (along with a dominant bass) leads to their characteristic sound. To automatically identify such patterns we introduce several descriptors from the song lyrics to represent different types of rhymes.

For the analysis of rhyme structures we do not rely on lexical word endings, but rather apply a more correct approach based on phonemes – the sounds, or groups thereof, in a language. Hence, we first need to transcribe the lyrics to a phonetic representation. The words 'sky' and 'lie', for instance, both end with the same phoneme /ai/. Phonetic transcription is language dependent, thus the language of song lyrics first needs to be identified, using e.g. the text categoriser 'TextCat' [5] to determine the correct transcripator to apply. However, for our test collections presented in this paper we considered only songs in English language, and we therefore exclusively use English phonemes. For the transcription step, we utilise the 'Analysing Sound Patterns' software package<sup>3</sup>. This package includes a phoneme transcripator, which is derived from early work on text-to-speech translation [9], which introduced a set of 329 letter-to-sound rules that translate from English text to the international phonetic alphabet (IPA).

After transcribing the lyrics into this phoneme representation, we distinguish two basic patterns of subsequent lines in a song text: *AA* and *AB*. The former represents

<sup>3</sup> <http://www2.eng.cam.ac.uk/~tpl/asp/>

two rhyming lines, while the latter denotes non-rhyming. Based on these basic patterns, we extract the features described in Table 2.

As the simplest structure, a ‘Couplet’ *AA* describes the rhyming of two or more subsequent pairs of lines. It usually occurs in the form of a ‘Clerihew’, i.e. several blocks of Couplets such as *AABBCC*. Another common pattern is the alternating rhyme, in the form of *ABAB*. An *enclosing rhyme*, defined as *ABBA*, denotes the rhyming of the first and fourth, as well as the second and third out of four lines. Based on these structure, we further measure ‘RhymePercent’, the percentage of lines with rhyming patterns versus the total number of lines in a song. Besides, we define the unique rhyme words as the fraction of unique terms used to build rhymes ‘UniqueRhymeWords’, which describes whether rhymes are frequently formed using the same word pairs, or a wide variety of words is used for the rhymes.

For our initial studies, we do not take into account rhyming schemes based on assonance, semirhymes, or alliterations. We also do not yet incorporate more elaborate rhyme patterns, especially not the less obvious ones, such as the ‘Ottava Rhyme’ of the form *ABABABCC*, and others. Also, we assign to all the rhyme forms the same weights, i.e. we for example do not give more importance to complex rhyme schemes. Experimental results lead to the conclusion that some of these patterns may well be worth studying. An experimental study on the frequency of occurrences might be a good starting point first, as modern popular music does not seem to contain many of these patterns.

## 4 Experiments

In this section we first introduce the test collections we use, followed by an illustration of some selected characteristics of our new features on these collections. We further present the results of our experiments, where we compare the performance of audio features and text features using various classifiers.

### 4.1 Test Collections

Music information retrieval research in general suffers from a lack of standardised benchmark collections, which is mainly attributable to copyright issues. Nonetheless, some collections have been used frequently in the literature, such as the collections provided for the ISMIR 2004 ‘rhythm’ and ‘genre’ contest tasks, or the collection presented in [37]. However, for the first two collections, hardly any lyrics are available, as they are either instrumental songs, or their lyrics were not published electronically. For the latter, no meta-data is available revealing the song titles, making the automatic fetching of lyrics impossible. The collection used in [14] turned out to be infeasible for our experiments. It consists of only about 260 pieces, and was not initially used for genre classification: it was compiled from only about 20



**Table 3** Composition of the two small (*collection\_600* and *collection\_660*) and two large (*collection\_3000* and *collection\_3120*) test collections

Genre	collection_600			collection_3000		
	Artists	Albums	Songs	Artists	Albums	Songs
Country	6	13	60	9	23	227
Folk	5	7	60	11	16	179
Grunge	8	14	60	9	17	181
Hip-Hop	15	18	60	21	34	380
Metal	22	37	60	25	46	371
Pop	24	37	60	26	53	371
Punk Rock	32	38	60	30	68	374
R&B	14	19	60	18	31	373
Reggae	12	24	60	16	36	181
Slow Rock	21	35	60	23	47	372
Total	159	241	600	188	370	3009
Genre	collection_660			collection_3120		
	Artists	Albums	Songs	Artists	Albums	Songs
Children’s music	7	5	60	7	5	109
Total	166	246	660	195	375	3118

different artists, and it was not well distributed over several genres (we specifically wanted to circumvent unintentionally classifying artists rather than genres).

To elude these limitations, we opted to compile our own test collections; more specifically, we first constructed two test collections different in size, first presented in [23]. For the first of these databases, we selected a total number of 600 songs (*collection\_600*) as a random sample from a private collection. We aimed at having a high number of different artists, represented by songs from different albums, in order to prevent biased results by too many songs from the same artist and album. This collection thus comprises songs from 159 different artists, stemming from 241 different albums. The ten genres listed in the left-hand side of Table 3 are represented by 60 songs each. Note that the number of different artists and albums is not equally spread, which is closer to a real-world scenario, though.

We then automatically fetched lyrics for this collection from the Internet using the lyrics scripts provided for the Amarok Music Player<sup>4</sup>. These scripts are simple wrappers for popular lyrics portals on the Web. To obtain all lyrics we used one script after another until all lyrics were available, regardless of the quality of the texts with respect to content or structure. Thus, the collection is named *collection\_600\_uncleansed*.

In order to evaluate the impact of proper lyrics preprocessing, we then manually cleansed the automatically collected lyrics. This is a tedious task, which first involves checking whether the fetched lyrics were matching the song at all. Then, we corrected the lyrics both in terms of structure and content, i.e. all lyrics were manually corrected in order to remove additional markup like ‘[2x]’, ‘[intro]’ or ‘[chorus]’, and to include the unabridged lyrics for all songs. We payed special at-

<sup>4</sup> <http://amarok.kde.org>

attention to completeness in terms of the resultant text documents being as adequate and proper transcriptions of the songs’ lyrics as possible. This collection, which differs from *collection\_600\_uncleansed* only in the song lyrics quality, is thus called *collection\_600\_cleansed*. Effects of manually cleansing lyrics as opposed to automatic crawling from the Web on the performance of the lyrics features, as well as the impact of stemming, were studied in [23] and [24]. As their impact has been found to be rather small, and not consistently improving or degrading the classification results, detailed studies on this issue are thus omitted here, and in the following experiments we only employ the cleansed version of the collection.

To evaluate our findings from the smaller test collection on a larger one, we constructed a more diversified database. This collection includes all the songs of the smaller collection, and consists of 3.010 songs, which can be seen as prototypical for a private collection. The numbers of songs per genre range from 179 in ‘Folk’ to 381 in ‘Hip-Hop’. Detailed figures about the composition of this collection can be taken from the right-hand side in Table 3. To be able to better relate and match the results obtained for the smaller collection, we only selected songs belonging to the same ten genres as in the *collection\_600*.

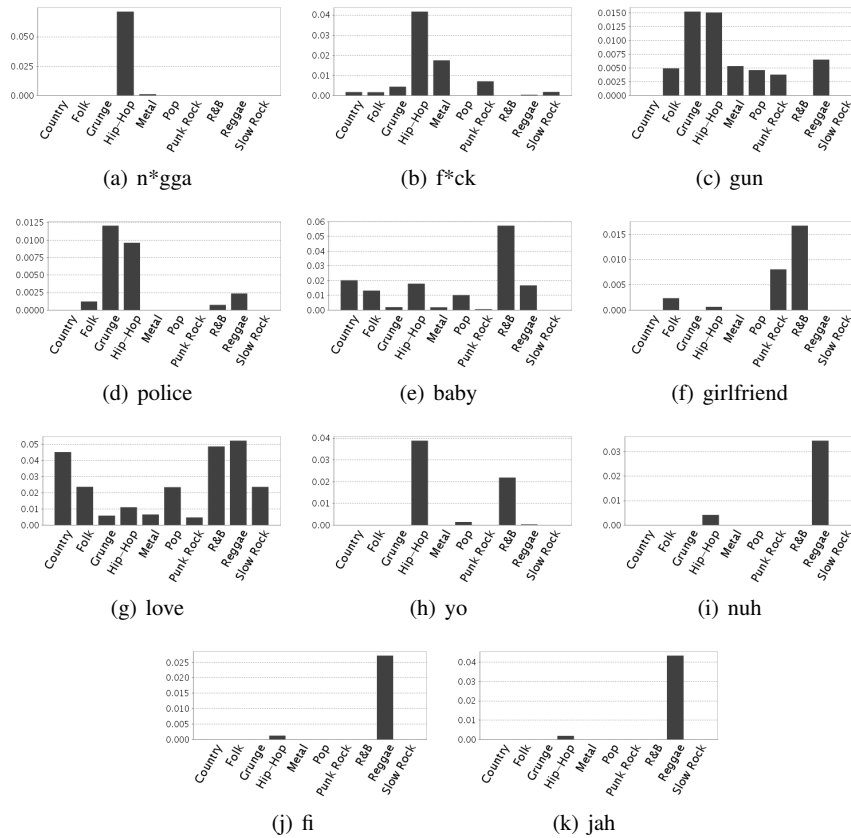
In a novel set of experiments, we added one more genre to these existing collections, namely children’s music, consisting of nursery rhymes and similar songs. The pieces of music in this genre in general have very distinctive acoustical properties, with a strong focus on vocals, and little instrumentation, which is often limited to the same instruments, such as guitars. Therefore, they already achieve high classification accuracies with audio-only features, and are thus an interesting challenge to test whether the lyrics features are able to improve performance also on genres that have distinctive acoustical properties. We therefore extended our smaller test database by 60 more songs, thus creating the new database *collection\_660*, and added a total of 109 songs to the larger collection, thus resulting in *collection\_3120*, both of which are illustrated also in Table 3.

## 4.2 Analysis of Selected Features

To demonstrate the ability of the newly proposed lyrics-based features to discriminate between different genres, we illustrate the distribution of the numerical values for these new features across the different genres. We focus on the most interesting features from each bag-of-words, rhyme, part-of-speech, and text statistic features, for the *collection\_600\_cleansed*.

First, plots for selected features from the bag-of-words set, all of which were among the highest ranked by the Information Gain feature selection method<sup>5</sup>, are presented in Figure 3. Of those high ranked terms, we selected some that have interesting characteristics regarding different classes. It can be generally said that no-

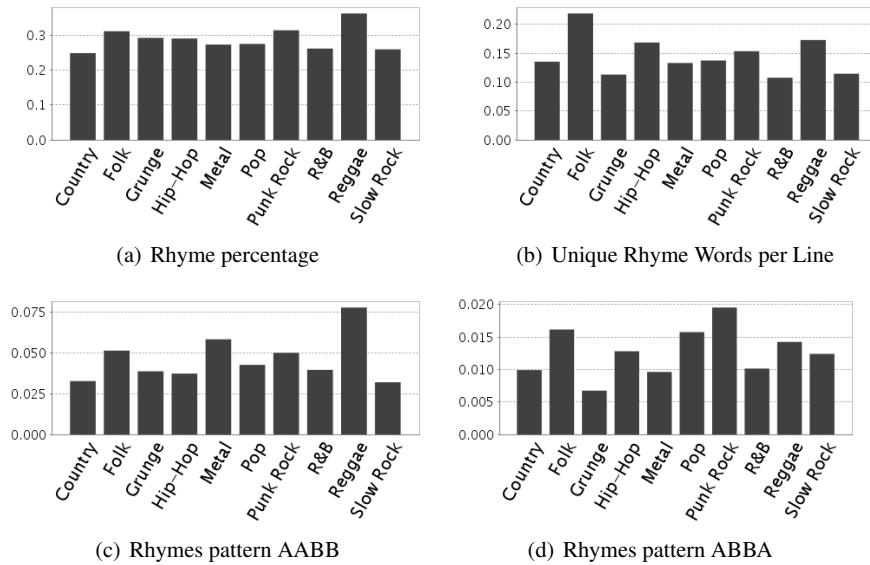
<sup>5</sup> Information Gain is a popular feature selection criterion, measuring the information obtained by a single term for category classification [31]



**Fig. 3** Average  $tf \times idf$  values of selected terms from the lyrics. Obscene words are obscured

tably ‘Hip-Hop’ seems to have a lot of commonly used terms, especially from swear and cursing language (subsequently obscured), or slang terms. This can be seen in Figure 3(a) and 3(b), showing the terms ‘n\*ggga’ and ‘f\*ck’. While ‘n\*ggga’ is used almost solely in ‘Hip-Hop’ (in many types – singular and plural forms, with ending ‘s’ and ‘z’), ‘f\*ck’ is also used in ‘Metal’ and to some lesser extent in ‘Punk-Rock’. On the contrary, ‘R&B’ and ‘Pop’ do not use the term at all, and other genres just very rarely employ it. Regarding the dominant topics, ‘Hip-Hop’ also frequently has *violence* and *crime* as content of their songs, which is exemplified in the terms ‘gun’ and ‘police’ in Figures 3(c) and 3(d), respectively. Both terms are also used in ‘Grunge’ and ‘Reggae’.

By contrast, ‘R&B’ has several songs concerning *relationships*, which is illustrated in Figures 3(e) and 3(f). Several genres deal with *love*, but to a very varying extent. In ‘Country’, ‘R&B’, and ‘Reggae’, this is a dominant topic, while it hardly occurs in ‘Grunge’, ‘Hip-Hop’, ‘Metal’ and ‘Punk-Rock’.



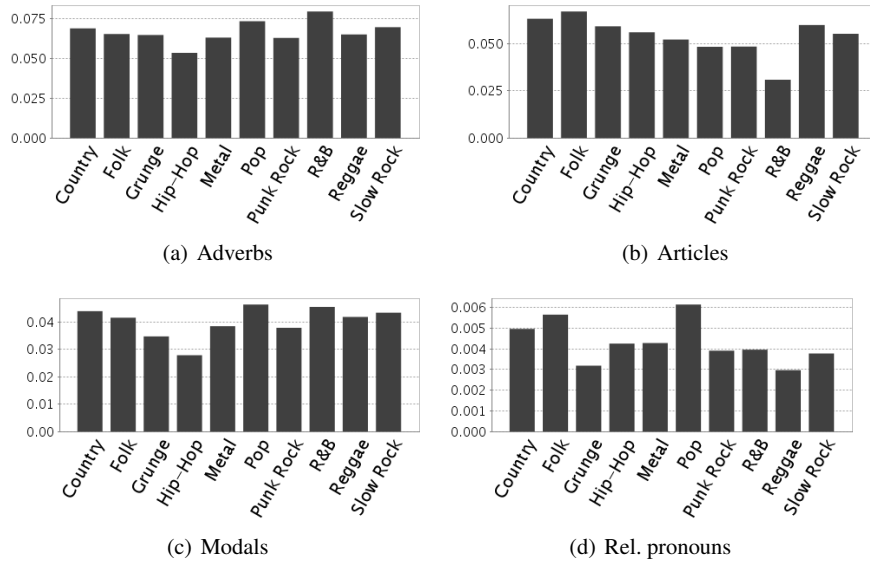
**Fig. 4** Average values for selected rhyme features

Another interesting aspect is the use of slang and colloquial terms, or more generally using a transcription of the phonetic sound of some words. This is especially used in the genres ‘Hip-Hop’ and ‘Reggae’, but also in ‘R&B’. Figure 3(h), for instance, shows that both ‘Hip-Hop’ and ‘R&B’ make use of the word ‘yo’, while ‘Reggae’ often uses a kind of phonetic transcription, as e.g. the word ‘nuh’ for ‘not’ or ‘no’, or many other examples, such as ‘mi’ (me), ‘dem’ (them), etc. ‘Reggae’ further employs a lot of particular terms, such as ‘jah’, which stands for ‘god’ in the Rastafari movement, or the Jamaican dialect word ‘fi’, which is used instead of ‘for’.

Summarising, a seemingly high amount of terms that are specific for ‘Hip-Hop’ and ‘Reggae’ can be observed, which should render those two genres well distinguishable from the others regarding bag-of-words features.

Figure 4 depicts selected rhyme features. ‘Reggae’ has the highest value of percentage of rhyming lines, while the other genres have rather equal usage of rhymes. ‘Folk’ may seem as using the most creative language for building those rhymes, which is manifested in the clearly higher number of unique words forming the rhymes, rather than repeatedly using the same words. ‘Grunge’ and ‘R&B’ seem to have distinctively lower values than the other genres. The distribution across the actual rhyme patterns used is also quite different over the genres, where ‘Reggae’ lyrics use a lot of *AABB* patterns, and ‘Punk Rock’ employs mostly *ABBA* patterns, while ‘Grunge’ makes particular little use of the latter.

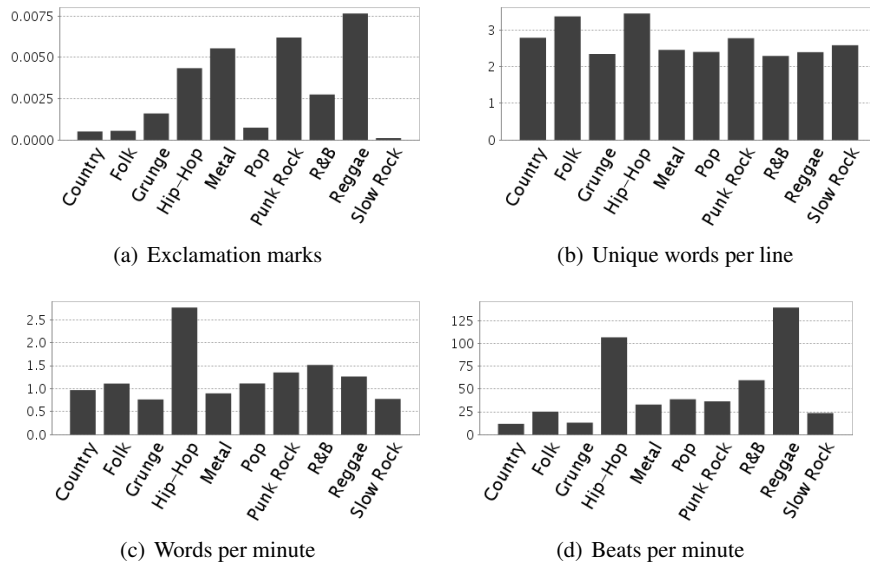
Figure 5 presents plots of the most relevant part-of-speech features. Adverbs seem to help discriminating ‘Hip-Hop’ with low and ‘Pop’ and ‘R&B’ with higher



**Fig. 5** Average values for selected part-of-speech features

values over the other classes. ‘R&B’ further can be well discriminated due to the infrequent usage of articles in the lyrics. Modals, on the other hand, are rarely used in ‘Hip-Hop’.

Finally, the most interesting features from the text statistics type are illustrated in Figure 6. ‘Reggae’, ‘Punk Rock’, ‘Metal’, and, to some extent, also ‘Hip-Hop’ seem to use very expressive language, which manifests in the higher percentage of exclamation marks appearing in the lyrics. ‘Hip-Hop’ and ‘Folk’ in general seem to have more creative lyrics, indicated by the higher percentage of unique words used as compared to other genres, which may have more repetitive lyrics. ‘Words per Minute’ appears to be a very good feature to distinguish ‘Hip-Hop’ as the genre with the fastest sung (or spoken) lyrics from music styles such as ‘Grunge’, ‘Metal’ and ‘Slow Rock’. The latter frequently have longer instrumental phases, especially longer lead-ins and fade-outs, and the pace of singing is adapted towards the general slower tempo of the (guitar) music. Comparing this feature with the well-known ‘Beats per Minute’ descriptor, it can be noted that the high tempo of ‘Hip-Hop’ lyrics coincides with the high number of beats per minute. ‘Reggae’ on the other hand has an even higher number of beats, and even though there are several pieces with fast lyrics, it is also characterised by longer instrumental passages, as well as words accentuated longer.



**Fig. 6** Average values for selected text statistic features and beats-per-minute

### 4.3 Experimental Results

After describing our experimental setup, we then discuss in detail the performance of the different audio and lyrics-only feature sets, and their combinations. We evaluate the impact of manually cleansing the lyrics, and specifically the performance of the newly added genre of children’s music.

#### 4.3.1 Setup

For each of the databases, we extract the audio and lyrics feature sets described in Section 3. We then build several combinations of these different feature sets, both separately within the audio and lyrics modalities, as well as combinations of audio and lyrics feature sets. This results in several dozens of different feature set combinations, out of which the most interesting ones are presented here. Most combinations with audio features are done with the SSD, as those are the best performing audio feature set.

For all our experiments, we employed the WEKA machine learning toolkit <sup>6</sup>, and unless otherwise noted used the default settings for the classifiers and tests. We utilised mainly k-Nearest-Neighbour, Naïve Bayes and Support Vector Machines. We performed the experiments based on a ten-fold cross-validation, which is fur-

<sup>6</sup> <http://www.cs.waikato.ac.nz/ml/weka/>

**Table 4** Classification accuracies and results of significance testing for various combinations of audio features for the 600 song collection (*collection\_600\_cleansed*). Statistically significant improvement or degradation over datasets (column-wise) is indicated by (+) or (–), respectively

Feature set	Dim.	4-NN	5-NN	NB	SVM/pol	SVM/lin	SVM/rbf
Chroma	48	18.33 -	18.50 -	18.77 -	19.60 -	22.53 -	14.63 -
MFCC	52	26.43 -	27.43 -	23.37 -	29.63 -	29.80 -	18.70 -
Marsyas	68	28.63 -	30.33 -	25.70 -	31.63 -	30.53 -	21.43 -
RP	1440	32.27 -	31.77 -	37.60 -	46.30 -	48.47 -	<b>44.20</b>
RH	60	29.73 -	29.03 -	31.13 -	36.03 -	36.47 -	28.97 -
SSD (base-line)	168	<b>48.97</b>	<b>49.57</b>	<b>44.57</b>	<b>56.63</b>	<b>59.37</b>	<b>44.20</b>
SSD / Chroma	216	50.70	<b>51.90</b>	42.37	59.30	59.17	43.13
SSD / Mars.	236	48.70	49.17	44.20	58.27	59.83	46.13
SSD / Mars. / Chroma	284	47.53	49.10	43.30	58.30	59.33	45.57
SSD / Mars. / Chroma / RH	344	47.73	48.60	42.67	59.67	60.90	46.97
SSD / Mars. / RH	296	49.50	49.63	43.90	<b>59.93</b>	<b>61.10</b>	47.67
SSD / MFCC	220	<b>51.23</b>	51.07	<b>44.73</b>	58.93	59.77	45.83
SSD / RH	228	49.37	49.80	43.17	58.57	60.37	46.83
SSD / RP	1608	41.77 -	39.87 -	41.77	57.73	60.23	52.87 +
SSD / RP / RH	1668	41.63 -	40.27 -	41.40	57.50	60.43	<b>53.30 +</b>

ther averaged over five repeated runs. All results given in this sections are micro-averaged classification accuracies. i.e. they are calculated giving equal weight to each document. Statistical significance testing is performed per column, using a paired t-test with an  $\alpha$  value of 0.05. In the following tables, plus signs (+) denote a significant improvement, whereas minus signs (–) denote significant degradation. The best results for each group of features are indicated by bold print.

#### 4.3.2 Small Database – Collection 600

Table 4 shows the results for genre classification experiments performed on the small collection using only audio-based feature sets. The columns show the results for three different types of machine learning algorithms, with different parameter settings:  $k$ -NN with  $k = 4$  and  $k = 5$  and employing Euclidean distance, Support Vector Machine with linear (SVM/lin), polynomial (quadratic, SVM/pol), and radial basis function (SVM/rbf) kernels, and a Naïve Bayes (NB) classifier. All six algorithm variations were applied to the six single feature sets, as well as nine different combinations thereof. Significance testing is performed per column, using the SSD features as the base line.

Generally, the highest classification results, sometimes by far better, are achieved with the SVM, which is thus the most interesting classifier for a more in-depth analysis. For the single audio feature sets, the Statistical Spectrum Descriptors (SSD) achieves the highest accuracy (59.37%) of all, followed by Rhythm Patterns (RP) with an accuracy of 48.47%, both with the SVM with linear kernel. SSD clearly outperforms all the other feature sets with statistical significance, except for the SVM

classifier with the RBF kernel, which achieves the exact same result on both SSD and RP. Apart from the Rhythm Patterns on the different SVM kernel variations, SSD features outperform the other sets by factors of 1.6 to 3.0.

Regarding the combinations of different audio feature sets, it was possible to increase the SSD baseline with some of the combinations with other feature sets, on one classifier with the Chroma features, on two in combination with MFCCs and by adding Marsyas and Rhythm Histograms (RH), and once by combining the SSD with the RH and RP. For most of those combinations, however, no change in performance that would be of statistical significance could be obtained; the only notable exception are the combination of SSD with RP only and both the RP and RH, yielding a significant degradation on the  $k$ -NN classifiers and a significant improvement utilising the SVM with RBF kernels. For the former, this most likely can be attributed to the generally poor performance of  $k$ -NN on high-dimensional feature sets, while for the latter, the baseline on the SSD-only features with the RBF kernel is very low compared to the other classifiers, even lower than  $k$ -NN.

Regarding the individual performance of the different classifiers, for  $k$ -NN it can be noted that there is no clear pattern on a better performance of a single classifier, even though the 5-NN seems to perform slightly better on most feature sets. For the SVMs, except for two out of the 15 feature sets, the linear kernel always got the highest results; especially the RBF kernel-based SVMs performed significantly worse. Thus, for the following experiments, we employ only the linear kernel.

After these initial experiments, we chose the highest result achievable with audio-only features, the SSD features, as the baseline we want to improve on. The SSDs show in general very good performance on our databases, with the achieved almost 60% clearly outperforming the minimal baseline of 10% on a database of ten equally-sized classes. Thus, they are as such a challenging baseline.

In Table 5, we detail the results of the different lyrics features, and their combination with the audio-only feature sets on the small, cleansed database (*collection\_600\_cleansed*), that is with automatic lyric fetching and manual checking of the retrieved lyrics. The experiments were again performed with six different classifiers, in contrary to those in Table 4 we employ also a 3-NN instead of the RBF kernel SVM, to give more details on the behaviour of the different  $k$  values. Indeed, 3-NN is the best performing of the  $k$ -NN family on a number of low-dimensional feature sets.

For the lyrics-only features, the rhyme features yield the lowest accuracies, while the Text-Statistics feature achieve a 29.73% accuracy, using a linear SVM. This result is remarkable, as it significantly outperforms the Chroma features, and is nearly achieving the results of MFCCs (0.07% short) and Marsyas (0.8% difference), coming at a very low dimensionality of only 23 features, while they are fast in computation. All combinations of the Text-Statistics features with the Part-of-Speech and / or Rhyme features achieve better results than Text-Statistics features alone.



**Table 5** Classification accuracies and results of significance testing for various combinations of lyrics features for the 600 song collection (*collection\_600\_cleansed*). Statistically significant improvement or degradation over datasets (column-wise) is indicated by (+) or (–), respectively

Feature set	Dim.	3-NN	4-NN	5-NN	NB	SVM/lin
Rhyme	6	13.93	13.20	13.37	15.00	13.77
POS	9	16.13	18.23	17.57	19.63	20.03
Text-stat (base line)	23	21.00	21.20	22.00	21.70	29.73
Text-stat / POS	32	<b>25.87</b> +	<b>25.17</b>	24.77	22.80	31.27
Text-stat / Rhyme	29	23.73	23.13	23.60	22.87	31.03
Text-stat / POS / Rhyme	38	22.90	24.47	<b>26.07</b>	24.20 +	30.63
Chroma (base-line)	48	17.87	18.33	18.50	18.77	22.53
Chroma / Text-stat	71	<b>23.07</b> +	<b>23.93</b> +	<b>23.87</b> +	22.33 +	32.87 +
Chroma / Text-stat / POS	80	21.43	21.00	21.57	22.53 +	34.87 +
Chroma / Text-stat / POS / Rhyme	86	21.47	20.83	21.53	<b>23.27</b> +	<b>35.07</b> +
Chroma / Text-stat / Rhyme	77	21.80	22.47	22.83	23.53 +	33.43 +
MFCC (base-line)	52	24.50	26.43	27.43	23.37	29.80
MFCC / Text-stat	75	27.83	<b>31.50</b> +	31.47	29.57 +	38.43 +
MFCC / Text-stat / POS	84	29.07 +	31.17 +	32.07	30.13 +	38.27 +
MFCC / Text-stat / POS / Rhyme	90	28.77	30.90	<b>32.50</b> +	<b>31.33</b> +	<b>39.63</b> +
MFCC / Text-stat / Rhyme	81	<b>29.53</b> +	30.87	31.40	29.90 +	38.50 +
MFCC / POS / Rhyme	67	23.37	26.20	28.10	26.53 +	34.53 +
Marsyas (base-line)	68	26.00	28.63	30.33	25.70	30.53
Mars. / Text-stat	91	29.23	30.60	32.90	30.50 +	37.83 +
Mars. / Text-stat / POS	100	29.57	<b>33.27</b> +	32.47	31.03 +	37.50 +
Mars. / Text-stat / POS / Rhyme	106	<b>30.30</b>	32.53	<b>34.10</b>	<b>31.83</b> +	<b>39.37</b> +
Mars. / Text-stat / Rhyme	97	28.97	32.37	33.73	30.90 +	39.00 +
SSD (base-line)	168	48.60	48.97	49.57	44.57	59.37
SSD / Text-stat	191	51.20	<b>53.07</b> +	<b>53.30</b> +	46.80	<b>64.53</b> +
SSD / Text-stat / POS	200	<b>51.97</b> +	51.00	51.70	46.73	64.07 +
SSD / Text-stat / POS / Rhyme	206	50.63	51.90	53.00	47.37 +	62.90 +
SSD / Text-stat / Rhyme	197	50.17	52.30 +	52.93	<b>47.57</b> +	63.93 +

When combining the different audio-features with the lyrics-based feature sets, it can be noted that in any combination, we achieve higher results than with the lyrics features alone. Especially for SVM, those improvements are always statistically significant when we include the Text-statistics features, which is also the case for all but two combinations when applying Naïve Bayes classification. For the  $k$ -NN, there is almost always one combination of features that leads to significant improvement. The combination of MFCCs is the only one where we can achieve *significant* improvement with adding just the Rhyme and POS features on SVM and NB, not using the Text-statistics features.

Compared to the baseline results achieved with SSDs, all four combinations of SSDs with the text statistic features yield higher performance, and at least one (and even all four in the case of SVMs) are statistically significant. The highest accuracy values are obtained for an SSD and text-statistic feature combination (64.53%),

**Table 6** Classification accuracies and results of significance testing for various combinations of lyrics features for the 660 song collection (*collection\_660*). Statistically significant improvement or degradation over the resp. audio-features-only baseline (column-wise) is indicated by (+) or (-), respectively

Feature set	Dim.	3-NN	4-NN	5-NN	NB	SVM/lin
Chroma	48	15.94	18.21	19.03	17.94	22.06
MFCC	52	25.39	26.55	27.67	24.27	30.06
Mars.	68	28.33	29.94	30.21	27.33	31.88
RH	60	27.82	29.27	28.76	30.55	36.42
RP	1440	28.12	30.67	30.55	37.06	48.70
SSD	168	<b>49.18</b>	<b>50.15</b>	<b>51.97</b>	<b>44.21</b>	<b>61.36</b>
BOW <sub>59</sub>	59	<b>15.24</b>	<b>15.85</b>	<b>15.06</b>	20.64	26.18
BOW <sub>150</sub>	150	10.97	10.24	9.42	24.97	29.52
BOW <sub>194</sub>	194	9.64	9.55	9.18	28.58	32.73
BOW <sub>653</sub>	653	11.21	10.03	10.03	<b>32.58</b>	<b>33.52</b>
BOW <sub>1797</sub>	1797	10.47	11.20	10.87	30.90	31.23
Mars. / Text-stat / POS	100	<b>32.55</b> +	<b>34.27</b> +	35.27 +	33.03 +	39.94 +
Mars. / Text-stat / Rhyme	97	30.73	32.67	34.39 +	33.06 +	41.33 +
Mars. / BOW <sub>248</sub>	316	26.06	25.73	26.42	28.91	41.67 +
Mars. / BOW <sub>653</sub>	721	17.00 -	19.18 -	21.64 -	33.27 +	42.61 +
Mars. / BOW <sub>194</sub> / Text-stat	285	31.33	33.30	<b>35.48</b> +	32.36 +	44.52 +
Mars. / BOW <sub>653</sub> / Text-stat	744	24.06 -	26.09 -	27.45	<b>34.45</b> +	<b>46.61</b> +
SSD / Text-stat	191	<b>53.42</b> +	54.06 +	<b>55.06</b> +	47.15 +	<b>66.27</b> +
SSD / Text-stat / Rhyme	197	<b>53.42</b> +	<b>54.12</b> +	54.91 +	<b>48.39</b> +	65.55 +
SSD / BOW <sub>14</sub>	182	48.03	50.85	50.76	47.06 +	58.88
SSD / BOW <sub>573</sub>	741	45.30	47.30	46.70 -	38.45 -	63.21
SSD / BOW <sub>385</sub> / Text-stat	576	50.67	53.76	54.88	38.00 -	65.30 +
SSD / BOW <sub>10</sub> / Text-stat / POS	210	49.33	51.36	53.97	48.24 +	62.09
SSD / BOW <sub>248</sub> / Text-stat / POS / Rhyme	454	50.85	53.88	53.06	37.39 -	65.70 +

which is 5.15%-points higher than the SSD-only value. It is interesting to note that adding part-of-speech and rhyme features does not help to improve on this result on SVMs, while it does on Naïve Bayes and 3-NN.

### 4.3.3 Small Database with Children’s Music – Collection 660

Table 6 illustrates the results of adding the additional genre of ‘Children’s music’ to the small collection, thus forming a database of 660 songs. First, it can be noted that, when compared to the results on the smaller collection, with SVM classification and linear kernel, the audio-only feature sets had mostly improved classification results (except Chroma and RH). The improvements range from 0.5% for RP to 2% for the SSD, which thus now achieves 61.36%. They are remarkable, as the classification task per-se has become a bit harder, with a minimal baseline of now 9.09%. The improvements thus already indicate that the new genre can be well captured

by audio-only features. Again, combinations with the rhyme and style features can improve the results significantly in many combinations.

On this database, we also present results from the bag-of-words features. In fact, we show a number of bag-of-words feature sets of different feature dimensions, which were obtained using different parameters for the document frequency thresholding based feature selection. Using this feature set alone, with a still moderate dimensionality of 653 topic terms, the best results are at around 33% for both SVM and Naïve Bayes. Notably,  $k$ -NN has rather poor performance, and further degrades with higher dimensionality. Also for the other classifiers it has to be noted that with a rising dimensionality, the accuracy starts to degrade again. Interestingly, both SVM and Naïve Bayes on BOW with 653 features can outperform the audio-only features Marsyas, MFCC and Chroma, most of it statistically significant, except for SVMs on the Marsyas feature set. Rhythm histograms are outperformed on the Naïve Bayes classifier, while Rhythm Patterns and SSD are significantly outperforming any of the bag-of-words features.

Also, it can be observed that adding the bag-of-words features can significantly improve the results obtained with the Marsyas features, even over the best combination of Marsyas with the rhyme and style features. Finally, adding bag-of-words to this aforementioned combination leads to a further improvement of more than 5%-points with SVM, thus totally more than 15%-points difference to the Marsyas-only features. Similar effects can be achieved for the other audio-only feature sets.

Regarding SSD features, the combination with the rhyme and style features again yields significant improvement on all classifiers. Combining them with the bag-of-words features can still yield better results than the SSD-only features, however, it leads to an improvement over the best combination with the rhyme and style features only on the Naïve Bayes classifier.

Finally, we want to examine the classification performance for each individual genre; for this, we train SVMs with a linear kernel on the SSD and the combination of SSD and Text-statistics feature set, which achieved the highest results. Table 7 gives the confusion matrix and the precision and recall values per class (in percent) for both feature sets, SSD on the left side, and SSD combined with Text-statistics on the right hand side.

With the audio features, high precision values can be achieved for the Children's music, R&B, Reggae, Punk Rock and Folk music, while Country, Slow Rock and especially Grunge perform poor.

When adding the Text-statistics features to the SSD features, eight out of eleven classes achieve a higher precision (of up to 25%), while the other three classes degrade in performance only by one percent; two out of those, namely Folk and R&B, however, gain 7% and 8%, resp, in recall. Overall, the average precision, as well as the recall and the F-measure<sup>7</sup>, thus rise from values around 61% to approximately 67%. The biggest increase in precision is achieved for Hip-Hop, which improves

---

<sup>7</sup> The F-Measure or F-score is a commonly used measure including both precision and recall. In our case, we specifically employ the  $F_1$ -measure, calculated as  $\frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$

**Table 7** Confusion matrix on the *collection\_660*: SSD (left) vs. SSD combined with Text-statistics (right). Precision and recall are measured per class

												classified as											
a	b	c	d	e	f	g	h	i	j	k	genre	a	b	c	d	e	f	g	h	i	j	k	
34	3	0	0	2	8	0	0	2	10	1	a = Country	38	5	0	0	2	6	0	0	1	8	0	
9	39	0	1	1	4	0	0	0	5	1	b = Folk	6	43	1	0	0	1	2	0	0	7	0	
0	2	47	0	1	4	1	0	1	4	0	c = Grunge	1	2	49	1	1	2	1	0	0	3	0	
0	2	0	39	0	3	1	6	8	0	1	d = Hip-Hop	0	0	0	50	0	0	1	6	2	0	1	
2	3	3	0	34	4	10	0	0	4	0	e = Metal	0	6	2	0	36	4	8	0	0	4	0	
10	3	9	4	4	11	3	2	1	11	2	f = Pop	9	3	8	0	4	16	3	4	0	12	1	
5	2	5	0	10	2	36	0	0	0	0	g = Punk Rock	4	2	6	0	9	1	37	0	0	1	0	
2	0	0	10	0	3	0	40	2	1	2	h = R&B	3	0	0	3	0	3	0	45	5	1	0	
0	1	0	7	0	1	0	2	45	0	4	i = Reggae	2	0	0	2	0	1	0	3	47	0	5	
8	1	8	1	3	5	1	1	1	27	4	j = Slow Rock	7	1	5	1	5	9	0	2	2	26	2	
1	0	0	0	0	1	0	1	3	2	52	k = Children's	0	1	0	0	0	0	0	0	3	1	55	
47	69	65	63	62	23	69	76	71	42	77	Precision	54	68	69	88	63	37	71	75	78	41	86	
57	65	78	65	57	18	6	67	75	45	87	Recall	63	72	82	83	6	27	62	75	78	43	92	

from 63% to 88%; much of this increase is likely to be attributed to the 'words-per-minute' feature. Other genres that improve greatly in precision are Reggae and Pop, even though the latter still has a low absolute precision. Of special interest is also the genre of Children's music. We noted before that children's music has some specific characteristics. This manifests e.g. in a focus on vocals, and a very limited set of instruments used, mainly guitars and pianos. Therefore, this genre can be well identified with audio-based feature sets, and indeed has a high recall of 52 out of 60, or 87%, and a high precision of 78% already with the SSD features. However, even in such cases, combining the audio features with lyrics-based features can improve the performance, in this specific case by raising the recall to 92%, and the precision to 86%, when combined with the Text-statistics features. The number of songs wrongly assigned into this genre greatly reduces, from 15 to only 8 songs.

#### 4.3.4 Large Database

To confirm our findings from the small database, we further performed experiments on the large collections (*collection\_3000*, *collection\_3120*). We again compare the results of the single audio and lyrics feature sets, and the combinations thereof. As there is not much difference in the variations of the  $k$ -NN algorithms, we now only present the results of the best-performing of the tested versions, 5-NN. For the SVMs, we again used a linear kernel.

The three centre columns in Table 8 give an overview of the accuracies of the different feature sets. For the audio-only features, we can observe an increased accuracy for most of the features set and classifier combinations, as compared to the smaller collection. In the case of the best-performing SSDs on SVMs, the increase

**Table 8** Classification accuracies and results of significance testing for the large collections. Statistically significant improvement or degradation over datasets (column-wise) is indicated by (+) or (–), respectively

Feature set	Dim.	<i>Collection_3000</i>			<i>Collection_3120</i>		
		NB	SVM	5-NN	NB	SVM	5-NN
Chroma	48	18.96	24.01	21.24	16.57	23.18	20.35
MFCC	52	27.64	33.57	31.27	26.55	32.66	31.21
Mars.	68	30.20	37.65	34.14	29.23	36.97	33.93
RP	1440	34.44	55.65	41.10	34.27	55.73	39.90
RH	60	29.26	35.05	34.44	29.03	34.17	33.86
SSD	168	<b>42.04</b>	<b>66.35</b>	<b>61.85</b>	<b>39.29</b>	<b>65.84</b>	<b>60.79</b>
Rhyme	6	16.68	16.11	16.97	16.57	16.09	17.70
POS	9	<b>23.67</b>	23.94	21.14	<b>23.60</b>	23.22	20.46
Text-stat	23	17.27	<b>28.70</b>	<b>24.78</b>	17.09	<b>28.16</b>	<b>24.79</b>
POS / Rhyme	15	<b>23.20+</b>	24.43-	21.66-	<b>23.27+</b>	24.55-	21.41-
Text-stat / POS	32	18.84+	31.23+	<b>25.91</b>	19.18+	31.41+	<b>25.72</b>
Text-stat / POS / Rhyme	38	20.15+	<b>31.24+</b>	25.10	20.88+	<b>31.64+</b>	25.47
Chroma / Text-stat	71	21.00+	32.61+	26.00+	20.64+	32.34+	25.97+
Chroma / Text-stat / POS	80	21.97+	35.95+	27.19+	22.10+	35.94+	26.44+
Chroma / Text-stat / POS / Rhyme	86	<b>22.31+</b>	<b>36.19+</b>	<b>27.35+</b>	<b>22.97+</b>	<b>36.54+</b>	<b>27.20+</b>
Chroma / Text-stat / Rhyme	77	21.53+	33.45+	26.06+	21.89+	33.45+	25.83+
Chroma / POS / Rhyme	63	21.48+	30.12+	25.19+	21.57+	30.01+	24.47+
MFCC / Text-stat	75	24.86-	40.48+	33.98+	24.35	40.40+	33.69+
MFCC / Text-stat / POS	84	26.03	41.71+	<b>35.50+</b>	26.21	41.76+	<b>34.94+</b>
MFCC / Text-stat / POS / Rhyme	90	26.83	<b>42.28+</b>	34.07+	27.55	<b>42.35+</b>	33.67+
MFCC / POS	61	<b>30.22+</b>	36.84+	32.52	29.42+	36.07+	31.80
MFCC / POS / Rhyme	67	30.13+	37.15+	32.10	<b>30.40+</b>	37.29+	31.44
Mars. / Text-stat	91	27.08-	43.44+	35.77+	27.13	43.44+	35.92+
Mars. / Text-stat / POS	100	28.33	44.98+	<b>37.17+</b>	28.50	44.91+	<b>36.32+</b>
Mars. / Text-stat / POS / Rhyme	106	29.70	<b>45.08+</b>	35.71	30.22	<b>45.38+</b>	36.11+
Mars. / POS / Rhyme	83	<b>32.67+</b>	41.54+	34.14	<b>32.91+</b>	41.82+	33.68
SSD / Text-stat	191	43.70+	68.57+	62.41	42.14+	68.38+	61.80
SSD / Text-stat / POS	200	44.29+	<b>68.91+</b>	<b>62.77</b>	42.86+	<b>68.94+</b>	61.44
SSD / Text-stat / POS / Rhyme	206	<b>44.51+</b>	68.35+	62.36	<b>43.44+</b>	68.36+	61.35
SSD / Text-stat / Rhyme	197	44.10+	68.00+	62.02	42.75+	68.01+	<b>61.81</b>

is of 7%-points to 66.35%. Similar patterns can be observed for the lyrics-based features, even though the flagship Text-statistics feature set achieves a 1% lower result on the SVM.

Also for the combination of the audio feature sets with the lyrics based features, a generally higher accuracy than on the smaller database can be noted, with total gains of 12.18% (Chroma), 8.71% (MFCCs), and 7.43% (Marsyas). The improvement over the SSD when combining them with the lyrics features is not as high as on the smaller collection – the accuracy raised to 68.91%, constituting an improvement of 2.55%-points, which is statistically significant. In general, it seems that the influence of part-of-speech and rhyme features is higher in this database, as they are more

often part of the highest-performing feature set combination than in the smaller collection.

The right columns in Table 8 finally show a summary of the results on the large database, extended by adding about 110 songs from the children’s music genre. The audio-only features generally perform a bit worse, between 0.1 and 0.8% when using SVMs, a bit more on some of the other classifiers. The same holds true for the rhyme and style features, though in their combinations among themselves, for some classifiers, the results are slightly, at most about 0.4%, higher than without the children’s music genre. Similarly, most of the combinations of audio and lyrics features perform slightly better on this database.

## 5 Beyond Audio and Lyrics

Much of today’s research in Music Information Retrieval is driven by audio-only genres, and classification of pieces of music therein. However, user studies have revealed that this narrow focus poses certain problems. For example, *semantic genres* such as Christmas songs or love-songs, cannot be adequately captured by audio features, as they might comprise musical genres – Christmas songs can actually be classical music, pop songs, or punk rock. Christian Rock is a genre that can virtually only be detected via the song texts. Similarly, pop music is a genre that is generally difficult to grasp with only acoustical features, as the common property of pop music is maybe more in the orientation towards being commercial music, rather than in musical characteristics. Thus, it is important to incorporate additional modalities as sources for features describing music. Such sources can e.g. be the song lyrics, album covers, social web data, etc.

In this paper, we thus presented a set of rhyme and style features for automatic lyrics processing, namely features to capture characteristics such as rhyme, parts-of-speech, and text statistics of song lyrics. We further combined these new feature sets with the standard bag-of-words features and well-known feature sets for acoustic analysis of digital audio tracks. To show the positive effects of feature combination on classification accuracies in musical genre classification, we performed experiments on two test collections. A smaller collection, consisting of 600 songs was manually edited and contains high quality unabridged lyrics. We then extended this database by adding songs from the children’s music genre, which are already well distinguishable on the audio features, and thus posed an interesting challenge on whether there could be further performance gains with this new dataset. We further compiled a larger test collection, comprising more than 3000 songs, which was again analysed in two flavours, with and without the children’s music. Using only automatically fetched lyrics, we achieved similar results in genre classification.

The most notable results reported in this paper are statistically significant improvements in musical genre classification. We outperformed both audio features alone as well as their combination with simple bag-of-words features. We conclude that combination of feature sets is beneficial in two ways: a) possible reduction in

dimensionality, and b) statistically significant improvements in classification accuracies. Future work hence is motivated by the promising results presented in this paper. Noteworthy future research areas in terms of machine learning are on more sophisticated ways of feature combination via ensemble classifiers, which pay special attention to the unique properties of single modalities and the different characteristics of certain genres in specific parts of the feature space. Additionally, a more comprehensive investigation of feature selection techniques and the impact of individual/global feature selection might further improve results.

Another topic for future research is the continued expansion of modalities and types of feature representations to be used for music analysis. A ‘glass-ceiling’ of achievable performance in regards to music information retrieval based on naïve timbral audio features only is discussed in [2]. It is further suggested that more high-level musical features are needed to overcome this limitation. While improved audio feature sets have been designed to address this issue, it is certainly of interest to look beyond the audio-only domain.

Steps in this direction have been discussed in this paper. Yet, we need to expand way beyond this scope. Album covers, for example, are carefully designed for specific target groups. Searching for music in a record shop is facilitated by browsing through album covers. There, album covers can, and have to, reveal very quickly the musical content of the album, and are thus used as strong visual clues [7]. Due to well-developed image recognition abilities of humans, this task can be performed very efficiently, much faster than listening to excerpts of the songs. Also, [4] suggests that ‘an essential part of human psychology is the ability to identify music, text, images or other information based on associations provided by contextual information of different media’. It further suggests that a well-chosen cover of a book can reveal it’s contents, or that lyrics of a familiar song can remind one of the song’s melody.

However, capturing the semantic meaning of album covers is a challenging task, requiring advanced pattern recognition and image retrieval methods. Concepts in the covers are more difficult to grasp than by simple colour histograms (even though for some genres, such as Gothic with a focus on dark/black colours, this feature might be a suitable candidate). More than that, it seems necessary to employ algorithms to detect the fonts used, face recognition to detect whether or not the singer or band feature on the cover, what scenery is depicted to e.g. indicate folk music, or which objects, instruments, etc. are present, down to understanding the sentiment and emotions of cover images.

This breath of information extends way beyond a cover, the song itself, or its recording. It encompasses cultural aspects and community feelings as expressed by subculture language, clothes and other aspects of social groupings.

Music may seem to be mono-modal, audio-only at first glance. Yet, it is inherently multimodal, living from, playing with and serving information on a multitude of layers. It needs to be appreciated and covered in all its multimodal complexities if we want to fully explore its richness and do justice to its versatility.

## References

1. Eric Allamanche, Jürgen Herre, Oliver Hellmuth, Bernhard Fröba, Thorsten Kastner, and Markus Cremer. Content-based identification of audio material using MPEG-7 low level description. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 197–204, Bloomington, IN, USA, October 15-17 2001.
2. 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.
3. Stephan Baumann, Tim Pohle, and Shankar Vembu. Towards a socio-cultural compatibility of MIR systems. In *Proceedings of the 5th International Conference of Music Information Retrieval (ISMIR'04)*, pages 460–465, Barcelona, Spain, October 10-14 2004.
4. Eric Brochu, Nando de Freitas, and Kejie Bao. The sound of an album cover: Probabilistic multimedia and IR. In Christopher M. Bishop and Brendan J. Frey, editors, *Proceedings of the 9th International Workshop on Artificial Intelligence and Statistics*, Key West, FL, USA, January 3-6 2003.
5. William B. Cavnar and John M. Trenkle. N-gram-based text categorization. In *Proceedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval (SDAIR'94)*, pages 161–175, Las Vegas, USA, 1994.
6. Holger Crysandt and Jens Wellhausen. Music classification with MPEG-7. In *Proceedings of SPIE-IS&T Electronic Imaging*, volume 5021 of *Storage and Retrieval for Media Databases*, pages 307–404, Santa Clara (CA), USA, January 2003. The International Society for Optical Engineering.
7. Sally Jo Cunningham, Nina Reeves, and Matthew Britland. An ethnographic study of music information seeking: implications for the design of a music digital library. In *Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries (JCDL '03)*, pages 5–16, Washington, DC, USA, 2003. IEEE Computer Society.
8. J. Stephen Downie. *Annual Review of Information Science and Technology*, volume 37, chapter Music Information Retrieval, pages 295–340. Information Today, Medford, NJ, 2003.
9. Honey S. Elovitz, Rodney Johnson, Astrid McHugh, and John E. Shore. Letter-to-sound rules for automatic translation of English text to phonetics. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 24(6):446–459, 1976.
10. Jonathan Foote. An overview of audio information retrieval. *Multimedia Systems*, 7(1):2–10, 1999.
11. M. Goto. A chorus section detection method for musical audio signals and its application to a music listening station. *IEEE Transactions on Audio, Speech & Language Processing*, 14(5):1783–1794, 2006.
12. Denny Iskandar, Ye Wang, Min-Yen Kan, and Haizhou Li. Syllabic level automatic synchronization of music signals and text lyrics. In *Proceedings of the ACM 14th International Conference on Multimedia (MM'06)*, pages 659–662, New York, NY, USA, 2006.
13. Peter Knees, Markus Schedl, Tim Pohle, and Gerhard Widmer. An Innovative Three-Dimensional User Interface for Exploring Music Collections Enriched with Meta-Information from the Web. In *Proceedings of the ACM 14th International Conference on Multimedia (MM'06)*, pages 17–24, Santa Barbara, California, USA, October 23-26 2006.
14. Peter Knees, Markus Schedl, and Gerhard Widmer. Multiple lyrics alignment: Automatic retrieval of song lyrics. In *Proceedings of 6th International Conference on Music Information Retrieval (ISMIR'05)*, pages 564–569, London, UK, September 11-15 2005.
15. Cyril Laurier, Jens Grivolla, and Perfecto Herrera. Multimodal music mood classification using audio and lyrics. In *Proceedings of the Seventh International Conference on Machine Learning and Applications (ICMLA'08)*, pages 688–693, San Diego, CA, USA, December 11–13 2008.
16. Tao Li, Mitsunori Ogihara, and Qi Li. A comparative study on content-based music genre classification. In *Proceedings of the International ACM Conference on Research and Development in Information Retrieval (SIGIR)*, pages 282 – 289, Toronto, Canada, 2003.



17. Thomas Lidy and Andreas Rauber. Evaluation of feature extractors and psycho-acoustic transformations for music genre classification. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR'05)*, pages 34–41, London, UK, September 11-15 2005.
18. Thomas Lidy, Andreas Rauber, Antonio Pertusa, and Jose Manuel Inesta. Improving genre classification by combination of audio and symbolic descriptors using a transcription system. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 61–66, Vienna, Austria, September 23-27 2007.
19. Beth Logan. Mel frequency cepstral coefficients for music modeling. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Plymouth, Massachusetts, USA, October 23-25 2000.
20. Beth Logan, Andrew Kositsky, and Pedro Moreno. Semantic analysis of song lyrics. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'04)*, pages 827–830, Taipei, Taiwan, June 27-30 2004.
21. Beth Logan and Ariel Salomon. A music similarity function based on signal analysis. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, Tokyo, Japan, August 2001.
22. Jose P. G. Mahedero, Álvaro Martínez, Pedro Cano, Markus Koppenberger, and Fabien Gouyon. Natural language processing of lyrics. In *Proceedings of the ACM 13th International Conference on Multimedia (MM'05)*, pages 475–478, New York, NY, USA, 2005.
23. Rudolf Mayer, Robert Neumayer, and Andreas Rauber. Combination of audio and lyrics features for genre classification in digital audio collections. In *Proceedings of the ACM Multimedia 2008*, pages 159–168. ACM New York, NY, USA, October 27-31 2008.
24. Rudolf Mayer, Robert Neumayer, and Andreas Rauber. Rhyme and style features for musical genre classification by song lyrics. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR'08)*, Philadelphia, PA, USA, September 14-18 2008.
25. Robert Neumayer and Andreas Rauber. Integration of text and audio features for genre classification in music information retrieval. In *Proceedings of the 29th European Conference on Information Retrieval (ECIR'07)*, pages 724–727, Rome, Italy, April 2-5 2007.
26. Robert Neumayer and Andreas Rauber. Multi-modal music information retrieval - visualisation and evaluation of clusterings by both audio and lyrics. In *Proceedings of the 8th Conference Recherche d'Information Assistée par Ordinateur (RIA0'07)*, Pittsburgh, PA, USA, May 29th - June 1 2007.
27. Nicola Orio. Music retrieval: A tutorial and review. *Foundations and Trends in Information Retrieval*, 1(1):1–90, September 2006.
28. Elias Pampalk, Arthur Flexer, and Gerhard Widmer. Hierarchical organization and description of music collections at the artist level. In *Research and Advanced Technology for Digital Libraries ECDL'05*, pages 37–48, 2005.
29. Elias Pampalk, Andreas Rauber, and Dieter Merkl. Content-based organization and visualization of music archives. In *Proceedings of ACM Multimedia 2002*, pages 570–579, Juan-les-Pins, France, December 1-6 2002.
30. Elias Pampalk, Andreas Rauber, and Dieter Merkl. Content-based Organization and Visualization of Music Archives. In *Proceedings of the ACM 10th International Conference on Multimedia (MM'02)*, pages 570–579, Juan les Pins, France, December 1-6 2002.
31. J. Ross Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.
32. Lawrence Rabiner and Biing-Hwang Juang. *Fundamentals of Speech Recognition*. Prentice Hall, 1993.
33. Andreas Rauber, Elias Pampalk, and Dieter Merkl. Using psycho-acoustic models and self-organizing maps to create a hierarchical structuring of music by musical styles. In *Proceedings of the 3rd International Symposium on Music Information Retrieval (ISMIR'02)*, pages 71–80, Paris, France, October 13-17 2002.
34. Andreas Rauber, Elias Pampalk, and Dieter Merkl. The SOM-enhanced JukeBox: Organization and visualization of music collections based on perceptual models. *Journal of New Music Research*, 32(2):193–210, June 2003.

35. Gerald Salton. *Automatic text processing – The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley Longman Publishing Co., Inc., 1989.
36. Roger N. Shepard. Circularity in judgments of relative pitch. *The Journal of the Acoustical Society of America*, 36(12):2346–2353, 1964.
37. George Tzanetakis and Perry Cook. Marsyas: A framework for audio analysis. *Organized Sound*, 4(30):169–175, 2000.
38. George Tzanetakis and Perry Cook. Sound analysis using MPEG compressed audio. In *Proceedings of the International Conference on Audio, Speech and Signal Processing (ICASSP)*, Istanbul, Turkey, 2000.
39. George Tzanetakis and Perry Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, July 2002.
40. Ziyong Xiong, Regunathan Radhakrishnan, Ajay Divakaran, and Thomas S. Huang. Comparing MFCC and MPEG-7 audio features for feature extraction, maximum likelihood HMM and entropic prior HMM for sports audio classification. In *Proceedings of the International Conference on Multimedia and Expo (ICME)*, 2003.
41. Dan Yang and WonSook Lee. Disambiguating music emotion using software agents. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Barcelona, Spain, October 2004.
42. Yongwei Zhu, Kai Chen, and Qibin Sun. Multimodal content-based structure analysis of karaoke music. In *Proceedings of the ACM 13th International Conference on Multimedia (MM'05)*, pages 638–647, Singapore, 2005.