Combination of Audio and Lyrics Features for Genre Classification in Digital Audio Collections

Rudolf Mayer[1], Robert Neumayer[1,2], and Andreas Rauber[1]

[1]Department of Software Technology and
Interactive Systems,
Vienna University of
Technology,
Favoritenstraße 9-11, 1040, Vienna, Austria
{mayer,rauber}@ifs.tuwien.ac.at

[2]Department of Computer and
Information Science,
Norwegian University of
Science and Technology,
Sem Sælands vei 7-9, 7491, Trondheim, Norway
neumayer@idi.ntnu.no

ABSTRACT

In many areas multimedia technology has made its way into mainstream. In the case of digital audio this is manifested in numerous online music stores having turned into profitable businesses. The widespread user adaption of digital audio both on home computers and mobile players show the size of this market. Thus, ways to automatically process and handle the growing size of private and commercial collections become increasingly important; along goes a need to make music interpretable by computers. The most obvious representation of audio files is their sound – there are, however, more ways of describing a song, for instance its lyrics, which describe songs in terms of content words. Lyrics of music may be orthogonal to its sound, and differ greatly from other texts regarding their (rhyme) structure. Consequently, the exploitation of these properties has potential for typical music information retrieval tasks such as musical genre classification; so far, there is a lack of means to efficiently combine these modalities. In this paper, we present findings from investigating advanced lyrics features such as the frequency of certain rhyme patterns, several parts-of-speech features, and statistic features such as words per minute (WPM). We further analyse in how far a combination of these features with existing acoustic feature sets can be exploited for genre classification and provide experiments on two test collections.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: General; H.3.3 [**Information Search and Retrieval**]: [retrieval models, search process, selection process]

General Terms

Measurement, Experimentation

Keywords

Lyrics processing, audio features, feature selection, supervised learning, genre classification, feature fusion

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, audio data itself is not limited soley 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 actual genres. Many traditional 'Christmas' songs were interpreted by modern artists and are heavily influenced by their style; 'Punk Rock' variations are no more uncommon than 'Hip-Hop' or 'Rap' versions. What all of these share, though, is a common set of topics to be sung about. These simple examples show that there is a whole level of semantics 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. However, text resources may not be found in the case of rather unknown artists or not available at all when dealing with 'Instrumental' tracks, for instance. Whether or not rhyming structures occur and the complexity of present patterns may be highly descriptive of certain genres. In some cases, for example when thinking about very 'ear-catching' songs, maybe even simple 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 lively nature of 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, 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 combination 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. Genre classification guarantees the comparability of different algorithms and feature sets. 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 to show the stability of our approach. We also perform classification experiments on automatically fetched lyrics in order to show in how far proper preprocessing contributes to the classification performance achieved for different feature sets.

This paper is strucured as follows. We start with giving an overview of previous relevant work in Section 2. We then give a detaild description of our approach and the advanced feature sets we use for analysing song lyrics and audio tracks alike; lyrics feature sets are detailed in Section 3. In Section 4 we apply our techniques to two audio corpora and provide 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 sub-area of information retrieval concerned with adequately accessing (digital) audio. Important research directions include, but are not limited to similarity retrieval, musical genre classification, or music analysis and knowledge representation. Comprehensive overviews of the research field are given in [4, 15]. The prevalent technique of processing audio files in information retrieval is to analyse the audio signal computed from plain wave files (or from other popular formats such as the MP3 or the lossless format Flac via a decoding step). Early experiments based on and an overview of content-based music information retrieval were reported in [5] as well as [20, 21], focussing on automatic genre classification of music. A well-known feature set for the abstract representation of audio is implemented in the Marsyas system [20]. In this work, we employ mainly the Rhythm Patterns and Statistical Spectrum Descriptors [9], which we will discuss in more detail in Section 3.1. Other feature sets may include for example MPEG-7 audio descriptors.

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 [11]. The definition of artist similarity via song lyrics is given in [10]. 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 lyrics alignment for automatic retrieval [8]. Lyrics are gathered by automatic alignment of the results obtained by Google queries. Preliminary results for genre classification using the rhyme features used later in this paper are reported in [12]; these results particularly showed that simple lyrics features may well be worthwile. 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 [22]. Some effort has also been spent on the automatic synchronisation of lyrics and audio tracks at a syllabic level [6]. A multi-modal approach to query music, text, and images with a special focus on album covers is presented in [2]. Other cultural data is included in the retrieval process e.g. in the form of textual artist or album reviews [1]. Cultural data is also used to provide a hierarchical organisations of music collections on the artist level in [16]. The system describes artists by terms gathered from web search engine results.

In [7], additional information like web data and album covers are used fo 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. The application of visualisation techniques for lyrics plus audio content based on (SOMs) is given in [14]. It demonstrates the potential of lyrics analysis for clustering collections of digital audio. Similarity of songs is visualised according to both modalities to compute quality measures with respect to the differences in distributions across clusterings in order to identify interesting genres and artists.

Experiments on the concatentaion of audio and bag-of-words features were reported in [13]. The results showed much potential for dimensionality reduction when using different types of features.

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 feature extraction described in the following. Finally, the results of both feature extraction processes are used for musical genre classification.

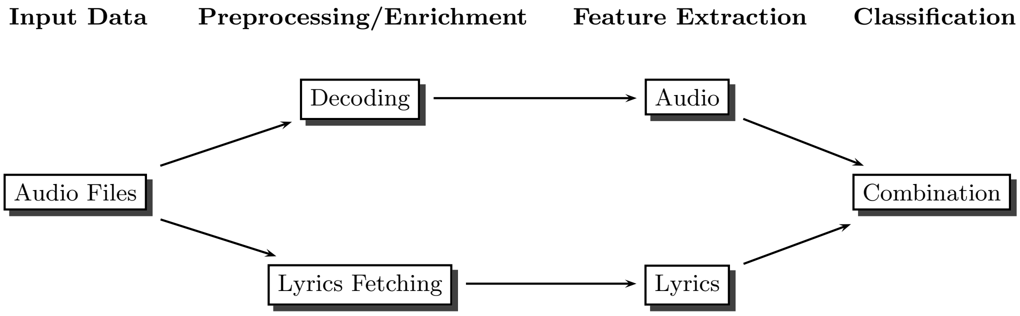


Figure 1: Processing architecture for combined audio and lyrics analysis stretching from a set of plain audio files to combined genre classification

3.1 Audio Features

In this section we describe the set of audio features we employed for our experiments, namely Rhythm Patterns, Statistical Spectrum Descriptors, and Rhythm Histograms. The latter two are based on the Rhythm Patterns features, and do skip or alter some of the processing steps, and result in a different feature dimensionality.

3.1.1 Rhythm Patterns

Rhythm Patterns (RP) are a feature set for handling audio data based on analysis of the spectral audio data and psycho-acoustic transformations [18], [9]. It has further been developed in the SOM-enhanced jukebox (SOMeJB) [17].

In a pre-processing stage, music in different file formats is converted to raw digital audio, and multiple channels are averaged to one. Then, the audio is split into segments of six seconds, possibly leaving out lead-in and fade-out segments. For example, for pieces of music with a typical duration of about 4 minutes, frequently the first and last one to four segments are skipped and out of the remaining segments every third one is processed.

The feature extraction process for a Rhythm Pattern is then composed of two stages. For each segment, the spectrogram of the audio is computed using the short time Fast Fourier Transform (STFT). The window size is set to 23 ms (1024 samples) and a Hanning window is applied using 50 % overlap between the windows. The Bark scale, a perceptual scale which groups frequencies to critical bands according to perceptive pitch regions [23], is applied to the spectrogram, aggregating it to 24 frequency bands. Then, the Bark scale spectrogram is transformed into the decibel scale, and further psycho-acoustic transformations are applied: computation of the Phon scale incorporates equal loudness curves, which account for the different perception of loudness at different frequencies [23]. Subsequently, the values are transformed into the unit Sone. The Sone scale 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 step, 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. After additional weighting and smoothing steps, a Rhythm Pattern exhibits magnitude of modulation for 60 modulation frequencies (between 0.17 and 10 Hz) on 24 bands, and has thus 1440 dimensions.

In order to summarise the characteristics of an entire piece of music, the feature vectors derived from its segments are simply averaged by computing the median. This approach extracts suitable characteristics of semantic structure for a given piece of music to be used for music similarity tasks.

3.1.2 Statistical Spectrum Descriptors

Computing Statistical Spectrum Descriptors (SSD) features relies on the first part of the algorithm for computing RP features. Statistical Spectrum Descriptors are based on the Bark-scale representation of the frequency spectrum. From this representation of perceived loudness a number of statistical measures is computed per critical band, in order to describe fluctuations within the critical bands. Mean, median, variance, skewness, kurtosis, min- and max-value are computed for each of the 24 bands, and a Statistical Spectrum Descriptor is extracted for each selected segment. The SSD feature vector for a piece of audio is then 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 – at matching performance in terms of genre classification accuracies [9].

3.1.3 Rhythm Histogram Features

The Rhythm Histogram features are a descriptor for rhythmical characteristics in a piece of audio. Contrary to the Rhythm Patterns and the Statistical Spectrum Descriptor, information is not stored per critical band. Rather, the magnitudes of each modulation frequency bin (at the end of the second phase of the RP calculation process) of all 24 critical bands are summed up, to form a histogram of 'rhythmic energy' per modulation frequency. The histogram contains 60 bins which reflect modulation frequency between 0.168 and 10 Hz. For a given piece of audio, the Rhythm Histogram feature set is calculated by taking the median of the histograms of every 6 second segment processed.

We further include the beats per minute (BPM) feature, computed as the modulation frequency of the peak of a Rhythm Histogram.

Table 1: 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

Table 2: Overview of text statistic features

Feature Name	Description
exclamation_mark, colon, single_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 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 or 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.

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.

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* [19], computed as:

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

This results in vectors of weight values for each document d in the collection, i.e. each lyrics document. This representation also introduces a concept of distance, as lyrics that contain a similar vocabulary are likely to be semantically related. We did not perform stemming in this setup, earlier experiments showed only negligible differences for stemmed and non-stemmed features (the rationale behind using non-stemmed terms is the occurrence of slang language in some genres).

3.2.2 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 [3] to determine the correct transcriptor. However, for our test collections presented in this paper we set the constraint to contain English songs only, and we therefore exclusively use English phonemes. Thus, we omit details on this step.

After transcribing the lyrics into a phoneme representation, we distinguish two patterns of subsequent lines in a song text: AA and AB. The former represents two rhyming lines, while the latter denotes non-rhyming. Based on these basic patterns, we extract the features described in Table 1.

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 $AABBCC$. $ABBA$, or *enclosing rhyme* denotes the rhyming of the first and fourth, as well as the second and third lines (out of four lines). We further measure 'RhymePercent', the percentage of rhyming blocks. Besides, we define the unique rhyme words as the fraction of unique terms used to build rhymes 'UniqueRhymeWords', which describes whether rhymes are frequentely formed using the same word pairs, or a wide variety of words is used for the rhymes.

In order to initially investigate the usefulness of rhyming at all, we do not take into account rhyming schemes based on assonance, semirhymes, alliterations, amongst others. We also did not yet incorporate more elaborate rhyme patterns, especially not the less obvious ones, such as the 'Ottava

Table 3: Composition of the small test collection (*collection_600*)

Genre	Artists	Albums	Songs
Country	6	13	60
Folk	5	7	60
Grunge	8	14	60
Hip-Hop	15	18	60
Metal	22	37	60
Pop	24	37	60
Punk Rock	32	38	60
R&B	14	19	60
Reggae	12	24	60
Slow Rock	21	35	60
Total	159	241	600

Table 4: Composition of the large test collection (*collection_3010*)

Genre	Artists	Albums	Songs
Country	9	23	227
Folk	11	16	179
Grunge	9	17	181
Hip-Hop	21	34	381
Metal	25	46	371
Pop	26	53	371
Punk Rock	30	68	374
R&B	18	31	373
Reggae	16	36	181
Slow Rock	23	47	372
Total	188	370	3010

Rhyme' of the form *ABABABCC*, and others. Also, we assign to all the rhyme forms the same weights, i.e. we do for example 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.

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. We count the 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 Text Statistic Features

Text documents can also be described by simple statistical measures based on word or character frequencies. Measures such as the average length of words or the ratio of unique words in the vocabulary might give an indication 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 2.

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[1].

4. EXPERIMENTS

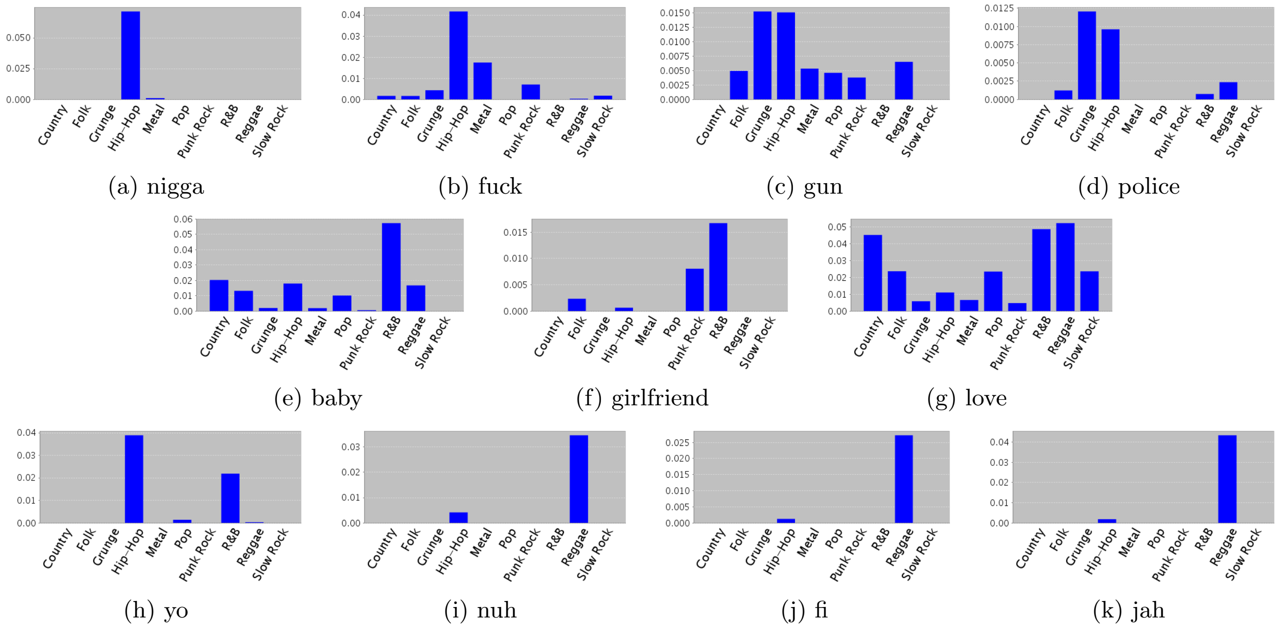
In this section we first introduce the test collections we used, followed by an illustration of some selected characteristics of our new features on these collections. We further present the results of our experiments, where we will compare the performance of audio features and text features using various classifiers. We put our focus on the evaluation of the smaller collection, and also investigate the effect of manually cleansing lyrics as opposed to automatic crawling off the Internet.

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 [20]. However, for the first two collections, hardly any lyrics are available as they are either instrumental songs or their lyrics were not published. For the latter, no ID3 meta-data is available revealing the song titles, making the automatic fetching of lyrics impossible. The collection used in [8] turned out to be infeasible for our experiments; it consists of about 260 pieces only and was not initially used for genre classification. Further, it was compiled from only about 20 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 to compile our own test collections; more specifically, we constructed two different test collections of differing size. For the first database, 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. This collection thus comprises songs from 159 different artists, stemming from 241 different albums. The ten genres listed in 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

[1] 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.



(a) nigga

(b) fuck

(c) gun

(d) police

(e) baby

(f) girlfriend

(g) love

(h) yo

(i) nuh

(j) fi

(k) jah

Figure 2: Average $tf \times idf$ values of selected terms from the lyrics

from the Internet using Amarok's[2] lyrics scripts. These scripts are simple wrappers for popular lyrics portals. 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_dirty*.

In order to evaluate the impact of proper preprocessing, we then manually cleansed the automatically collected lyrics. This is a tedious task, as it 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 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_dirty* only in the song lyrics quality, is thus called *collection_600_cleansed*.

Finally, we wanted to evaluate our findings from the smaller test collection on a larger, more diversified database of medium- to large-scale. This collection consists of 3.010 songs and can be seen as proto-typical 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 Table 4. 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*.

4.2 Analysis of Selected Features

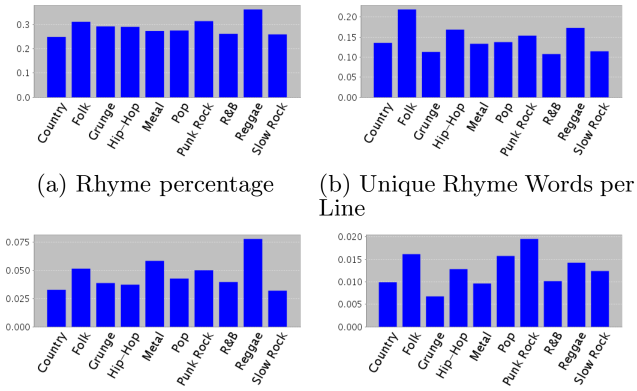
To investigate the ability of the newly proposed text-based features to discriminate between different genres, we illustrate the distribution of the values for these new features across the whole feature set. Due to space limitations, we focus on the most interesting features from each bag-of-words, rhyme, part-of-speech, and text statistic features; we also only show results for the *collection_600_cleansed*.

To begin with, we present plots for some selected features from the bag-of-words set in Figure 2. The features were all within the highest ranked by the Information Gain feature selection algorithm. Of those, we selected some that have interesting characteristics regarding different classes. It can be generally said that notably 'Hip-Hop' seems to have a lot of repeating terms, especially terms from swear and cursing language, or slang terms. This can be seen in Figure 2(a) and 2(b), showing the terms 'nigga' and 'fuck'. Whilst 'nigga' is almost solely used in 'Hip-Hop' (in many variations of singular and plural forms with ending 's' and 'z'), 'fuck' is also used in 'Metal' and to some extent in 'Punk-Rock'. By contrast, 'Pop' and 'R&B' do not use the term at all, and other genres just very rarely employ it. Topic wise, 'Hip-Hop' also frequentely has *violence* and *crime* as content of their songs, which is shown in in Figures 2(c) and 2(d), giving distribution statistics on the terms 'gun' and 'police'. Both terms are also used in 'Grunge' and to a lesser extent in 'Reggae'.

On the contrary, 'R&B' has several songs focusing on *relationships* as the topic, which is illustrated in Figures 2(e) and 2(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'.

Another interesting aspect is the use of slang and colloquial terms, or generally a way of transcribing the phonetic sound of some words to letters. This is especially used in the genres 'Hip-Hop' and 'Reggae', but also in 'R&B'. Figure 2(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. Also, 'Reggae' employs a lot of special terms,

[2] `http://amarok.kde.org`



(a) Rhyme percentage (b) Unique Rhyme Words per Line

(c) Rhymes pattern AABB (d) Rhymes pattern ABBA

Figure 3: Average values for selected rhyme features

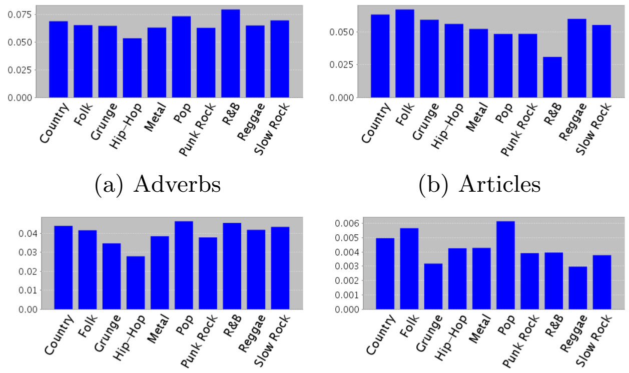
such es ‘jah’, which stands for ’god’ in the Rastafari movement, or the Jamaican dialect word ‘fi’, which is used instead of ‘for’.

It generally can be noted that there seems to be a high amount of terms that are specific for ‘Hip-Hop’ and ‘Reggae’, which should especially make those two genres well distinguishable from the others.

In Figure 3, some of the rhyme features are depicted. Regarding the percentage of rhyming lines, ‘Reggae’ has the highest value, while the other genres have rather equal usage of rhymes. However, ‘Folk’ seems to use the most creative language for building those rhymes, which is manifested in the clearly higher number of unique words forming the rhymes, rather than repeating them. ‘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 4 shows plots of the most relevant of the the part-of-speech features. Adverbs seem to help discriminating ‘Hip-Hop’ with low and ‘Pop’ and ‘R&B’ with higher 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’.

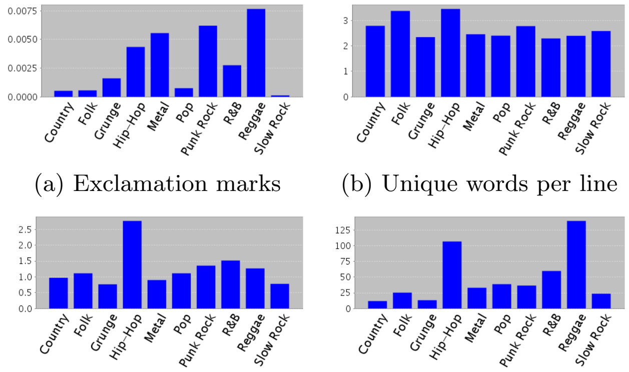
Some interesting features from the text statistics type are illustrated in Figure 5. ‘Reggae’, ‘Punk Rock’, ‘Metal’, and, to some extent, also ‘Hip-Hop’ seem to use very expressive language; this manifests in the higher percentage of exclamation marks appearing in the lyrics. ‘Hip-Hop’ and ‘Folk’ seem to have more creative lyrics in general, as the percentage of unique words used is higher than in other genres which may have more repetitive lyrics. Finally, ‘Words per Minute’ is 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 are often characterised by having longer instrumental phases, especially longer lead-ins and fade-outs, as well as adapting the speed of the singing towards the general slower speed of the (guitar) music. To compare this feature with the well-known ‘Beats per Minute’, it can be noted that the high tempo of ‘Hip-Hop’ lyrics coincides with the high number of beats per minute. ‘Reggae’ 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.



(a) Adverbs (b) Articles

(c) Modals (d) Rel. pronouns

Figure 4: Average values for selected part-of-speech features



(a) Exclamation marks (b) Unique words per line

(c) Words per minute (d) Beats per minute

Figure 5: Average values for selected text statistic features and beats-per-minute

4.3 Experimental Results

All results given are micro-averaged classification accuracies. For significance we used a paired t-test, with α=0.05.

Table 5 shows results for genre classification experiments performed on the small collection with automatic lyric fetching (*collection_600_dirty*), i.e. no manual checking of retrieved lyrics. The columns show the results for three different machine learning algorithms: k-NN with $k = 3$, Support Vector Machine with polynomial kernel ($C = 1, exponent = 2$), and a Naïve Bayes classifier. All three algorithms were applied to 25 different combinations of the feature sets described in this paper. We chose the highest result achievable with audio-only features, the SSD features, as the baseline we want to improve on (SSDs show very good performance and as such are a difficult baseline). Significance testing is performed per row, i.e. the SSD features as base data and the results therewith are thus given in the first row of the table. Plus signs (+) denote a significant improvement, whereas minus signs (−) denote significant degradation.

Regarding ‘single-feature’ data sets, the SSD, classified with the SVM classifier, achieves the highest accuracy (59.17%)

Table 5: Classification accuracies and results of significance testing for the 600 song collection (*collection_600_dirty*). Statistically significant improvement or degradation over datasets (column-wise) is indicated by (+) or (−), respectively (paired t-test, α=0.05, micro-averaged accuracy)

Exp.	Dataset	Dim.	3-NN	SVM	NB
18	ssd (base classifier)	168	**49.29**	59.17	45.08
1	textstatistic	23	24.04 −	28.92 −	22.12 −
2	textstatisticpos	32	26.42 −	31.54 −	23.13 −
3	textstatisticposrhyme	38	25.54 −	31.25 −	25.37 −
4	textstatisticrhyme	29	25.17 −	28.58 −	24.17 −
5	lyricsssd	9608	23.42 −	54.83 −	37.71 −
6	lyricsssdtextstatistic	9631	24.33 −	56.33	37.08 −
7	lyricsssdtextstatisticpos	9640	24.25 −	56.54	37.37 −
8	lyricsssdtextstatisticposrhyme	9646	23.88 −	57.04	37.63 −
9	lyricsssdtextstatisticrhyme	9637	23.75 −	56.83	37.46 −
10	lyricsssdpos	9617	23.87 −	55.63	37.75 −
11	lyricsssdrh	9668	24.42 −	56.71	38.33 −
12	lyricsssdrhyme	9614	24.54 −	55.63	37.58 −
13	pos	9	18.21 −	21.83 −	21.88 −
14	posrhyme	15	19.37 −	24.13 −	23.54 −
15	rh	60	30.00 −	35.37 −	31.25 −
16	rhyme	6	14.87 −	16.42 −	16.92 −
17	rp	1440	30.04 −	48.37 −	36.96 −
19	ssdtextstatistic	191	49.04	**63.50** +	45.54
20	ssdtextstatisticpos	200	49.54	62.13 +	45.75
21	ssdtextstatisticrhyme	197	48.00	63.00 +	46.38
22	ssdpos	177	48.04	58.17	45.04
23	ssdposrhyme	183	46.96	58.08	46.08
24	ssdposrhymetextstatistic	206	49.04	62.04	**46.54**
25	ssdrhyme	174	47.46	58.67	45.79

of all, followed by RP with an accuracy of 48.37%. 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. Compared to the baseline results achieved with SSDs, all four combinations of SSDs with the text statistic features yield higher results when classified with SVMs, three of which are statistically significant. The highest accuracy values are obtained for a SSD and text-statistic feature combination (63.50%). It is interesting to note that adding part-of-speech and rhyme features does not help to improve on this result.

Using 3-NN, the highest values are achieved with SSD features alone, while the combinations with the new features yield slightly worse results, which are not significantly lower, though. With Naïve Bayes, the highest accuracy was achieved with a combination of SSD with part-of-speech, rhyme and text statistic features; again, this result was not statistically different too the base line.

Table 6 shows the same experiments performed on the manually cleansed version (*collection_600_cleansed*) of the same collection. The base data set remains identical (SSD). Overall, these experiments show similar results. It is notable, however, that accuracies for the best data sets are a bit higher than the ones achieved for the uncleansed collection. Again, the best result are achieved by SVM, but the highest overall accuracy values are this time obtained with the SSD-text statistic-POS feature set combination (64.50%, compared to a maximum of 63.50% before). This shows that lyrics preprocessing and cleansing can potentially lead to better detection rates for parts-of-speech and in turn may improve accuracy. In this set of experiments, the combination of SSD and all of our proposed feature sets shows noteable improvements too, three out of four statistically significant, pointing out that the higher the quality of the lyrics after preprocessing the better the performance of the additional features. Again, however, rhyme features seem to have the lowest impact of the three feature sets.

The other classifiers produce generally worse results than SVMs, but this time, the highest results were all in combination of SSDs either with text statistic features (k-NN) or with text statistic and rhyme features (Naïve Bayes). Even though it still is not statistically significant, the improvements are around 3% higher than the base line, and thus much bigger than in the uncleansed corpus. This is another clue that the new features benefit from improved lyrics quality by better preprocessing.

We also performed experiments on the large collection *collection_3010*; results are given in Table 7. Due to the fact that SVMs vastly outperformed the other machine learning algorithms on the small collection, we omitted results for k-NN and Naïve Bayes for the large collection. Again, we compare the highest accuracy in audio achieved with the SSD feature set to the combination of audio features and our style features. Even though the increases seem to be smaller than with the *collection_600* – largely due to the lower effort spent on preprocessing of the data – we still find statistically significant improvements. All combinations of text statistic features with SSDs (experiments 11, 12, 13, and 16) perform significantly better. Combination experiments of SSDs and Lyrics features (experiments 18 and 19) achieved better rates than the SSD baseline, albeit not statistically significant. The dimensionality of these feature combinations, however, is much higher. Also, the document frequency thresholding we performed might not be the best way of feature selection.

Accuracies for all experiments might be improved by employing ensemble methods which are able to better take into account the unique properties of all single modality; different audio feature sets or combinations thereof might further improve results. Also, better techniques for feature selection

Table 6: Classification accuracies and significance testing for the 600 song collection (*collection_600_cleansed*). Statistically significant improvement or degradation over datasets (column-wise) is indicated by (+) or (−), respectively (paired t-test, α=0.05, micro-averaged accuracy)

Exp.	Dataset	Dim.	3-NN	SVM	NB
18	ssd (base classifier)	168	49.29	59.17	45.08
1	textstatistic	23	20.87 −	29.83 −	21.50 −
2	textstatisticpos	32	25.83 −	31.29 −	22.33 −
3	textstatisticposrhyme	38	24.00 −	30.88 −	24.21 −
4	textstatisticrhyme	29	23.08 −	30.96 −	23.25 −
5	lyricsssd	9434	22.58 −	53.46 −	37.62 −
6	lyricsssdtextstatistic	9457	22.50 −	55.12 −	37.42 −
7	lyricsssdtextstatisticpos	9466	22.42 −	54.33 −	36.96 −
8	lyricsssdtextstatisticposrhyme	9472	21.96 −	54.21 −	37.00 −
9	lyricsssdtextstatisticrhyme	9463	22.46 −	54.79 −	37.29 −
10	lyricsssdpos	9443	21.96 −	53.46 −	37.29 −
11	lyricsssdrh	9494	23.92 −	56.04 −	38.08 −
12	lyricsssdrhyme	9440	22.29 −	53.71 −	37.29 −
13	pos	9	16.33 −	19.21 −	19.71 −
14	posrhyme	15	17.46 −	21.38 −	21.17 −
15	rh	60	30.00 −	35.37 −	31.25 −
16	rhyme	6	14.37 −	14.46 −	14.75 −
17	rp	1440	30.04 −	48.37 −	36.96 −
19	ssdtextstatistic	191	51.71	64.33 +	47.79
20	ssdtextstatisticpos	200	**52.25**	**64.50** +	47.25
21	ssdtextstatisticrhyme	197	50.08	63.71	**48.21**
22	ssdpos	177	47.58	58.87	44.96
23	ssdposrhyme	183	47.54	58.50	45.75
24	ssdposrhymetextstatistic	206	50.63	63.75 +	47.42
25	ssdrhyme	174	47.75	58.62	45.46

based on, e.g., information theory and applied to multiple sets of features might lead to better results.

Table 7: Classification accuracies and results of significance testing for the 3010 song collection (non-stemming). Statistically significant improvement or degradation over different feature set combinations (column-wise) is indicated by (+) or (−), respectively

Exp.	Dataset	Dim.	SVM	
10	ssd	168	66.32	
1	textstatistic	23	28.72	−
2	textstatisticpos	32	28.72	−
3	textstatisticposrhyme	38	28.56	−
4	textstatisticrhyme	29	28.56	−
5	pos	9	12.66	−
6	posrhyme	15	15.83	−
7	rh-	60	35.01	−
8	rhyme	6	15.83	−
9	rp	1440	55.37	−
11	ssdtextstatistic	191	**68.72**	+
12	ssdtextstatisticpos	200	**68.72**	+
13	ssdtextstatisticrhyme-	197	68.16	+
14	ssdpos	177	66.32	
15	ssdposrhyme	183	66.38	
16	ssdposrhymetextstatistic	206	68.09	+
17	ssdrhyme	174	66.38	
18	lyricsssd	2140	66.44	
19	lyricsssdtextstatisticposrhyme	2178	67.06	

5. CONCLUSIONS

In this paper we presented a novel set of style features for automatic lyrics processing. We presented features to capture rhyme, parts-of-speech, and text statistics characteristics for 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 unabriged lyrics. To have comparison figures with automatic lyrics fetching from the internet, we also performed the same set of experiments on non-cleansed lyrics data. We further compiled a larger test collection, comprising more than 3010 songs. 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 in this paper. Noteworthy research areas are two-fold: (1) 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; and (2) improved ways of lyrics retrieval and preprocessing, as we showed its positive effect on classification accuracies. Additionally, a more comprehensive investigation of feature selection techniques and the impact of individual/global feature selection might further improve results.

Another interesting observation, though not the main intention of the experiments carried out, is that the Statistical Spectrum Descriptors significantly outperform the Rhythm Patterns. On the *collection_600*, the increase is from 48.37%

to 59.17%. On the *collection_3010*, the performance increase is from 55.37% to 66.32%. These results stand a bit in contrast to previous surveys, which saw SSD being sometimes marginally better or worse compared to Rhythm Patterns, and the major benefit of them thus being the great reduction of dimensionality from 1,440 to 168 features. It would hence be worth investigating the performance of these two feature sets on other collections as well; we particularly want to point out that the lack of publicly available test collection inhibits collaboration and evaluation in lyrics analysis.

6. REFERENCES

[1] S. Baumann, T. Pohle, and S. 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.

[2] E. Brochu, N. de Freitas, and K. Bao. The sound of an album cover: Probabilistic multimedia and IR. In C. M. Bishop and B. J. Frey, editors, *Proceedings of the 9th International Workshop on Artificial Intelligence and Statistics*, Key West, FL, USA, January 3-6 2003.

[3] W. B. Cavnar and J. 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.

[4] J. Downie. *Annual Review of Information Science and Technology*, volume 37, chapter Music Information Retrieval, pages 295–340. Information Today, Medford, NJ, 2003.

[5] J. Foote. An overview of audio information retrieval. *Multimedia Systems*, 7(1):2–10, 1999.

[6] D. Iskandar, Y. Wang, M.-Y. Kan, and H. 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. ACM.

[7] P. Knees, M. Schedl, T. Pohle, and G. 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.

[8] P. Knees, M. Schedl, and G. 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.

[9] T. Lidy and A. 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.

[10] B. Logan, A. Kositsky, and P. Moreno. Semantic analysis of song lyrics. In *Proceedings of the 2004 IEEE International Conference on Multimedia and Expo (ICME'04)*, pages 827–830, Taipei, Taiwan, June 27-30 2004.

[11] J. P. G. Mahedero, Á. Martínez, P. Cano, M. Koppenberger, and F. 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. ACM Press.

[12] R. Mayer, R. Neumayer, and A. 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. Accepted for publication.

[13] R. Neumayer and A. 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.

[14] R. Neumayer and A. 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 (RIAO'07)*, Pittsburgh, PA, USA, May 29th - June 1 2007. ACM.

[15] N. Orio. Music retrieval: A tutorial and review. *Foundations and Trends in Information Retrieval*, 1(1):1–90, September 2006.

[16] E. Pampalk, A. Flexer, and G. 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.

[17] E. Pampalk, A. Rauber, and D. 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. ACM.

[18] A. Rauber, E. Pampalk, and D. 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.

[19] G. Salton. *Automatic text processing – The Transformation, Analysis, and Retrieval of Information by Computer.* Addison-Wesley Longman Publishing Co., Inc., 1989.

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

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

[22] Y. Zhu, K. Chen, and Q. 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. ACM.

[23] E. Zwicker and H. Fastl. *Psychoacoustics, Facts and Models*, volume 22 of *Series of Information Sciences.* Springer, Berlin, 2 edition, 1999.