JOHANNES KEPLER
UNIVERSITÄT LINZ | JKU

Institut für
Computational
Perception

Automatically Extracting, Analyzing, and Visualizing Information on Music Artists from the World Wide Web

Dissertation zur Erlangung des akademischen Grades
Doktor der technischen Wissenschaften

Angefertigt am:

Institut für Computational Perception

Eingereicht von:

Dipl.Ing. Markus Schedl

Betreuung:

Univ.Prof. Dipl.Ing. Dr. Gerhard Widmer

Beurteilung:

Univ.Prof. Dipl.Ing. Dr. Gerhard Widmer
Ao.Univ.Prof. Dipl.Ing. Dr. Andreas Rauber

Linz, Juni 2008

Johannes Kepler Universität Linz
A-4040 Linz • Altenberger Straße 69 • http://www.jku.at • DVR 0093696

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Dissertation selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Kurzfassung

Im Rahmen dieser Dissertation wurden Methoden zur automatischen Gewinnung musikbezogener Informationen aus dem World Wide Web erarbeitet, implementiert und analysiert. Die Bereitstellung solcher Informationen wird in Zeiten des digitalen Musikvertriebes immer bedeutender, da die Nutzer von Online-Musikgeschäften sich über die reine Musikdatei hinausgehende Informationen erwarten. Zur Extraktion von Informationen über Musikkünstler und Bands aus dem Web wurden sowohl neue Techniken entwickelt, als auch existierende verfeinert. Diese Techniken sind den Wissenschaftsdisziplinen *music information retrieval*, *Web mining* und *information visualization* zuzuordnen. Konkret wurden auf Webseiten, die in Relation zu einem Musikkünstler oder einer Band stehen, Methoden des *Web content mining* angewandt um folgende Informationen zu gewinnen:

- Ähnlichkeiten zwischen Musikkünstlern oder Bands
- Prototypizität eines Künstlers oder einer Band für ein Genre
- beschreibende Eigenschaften eines Künstlers oder einer Band
- Bandmitglieder und Instrumentierung
- Bilder von Albencover

Hierfür wurden unterschiedliche Ansätze zur Extraktion dieser Informationen entwickelt und ausgiebig auf einer Vielzahl von Musiksammlungen evaluiert. Die Ergebnisse dieser Evaluierungsprozesse werden präsentiert. Außerdem entstanden im Rahmen dieser Dissertation Visualisierungsmethoden und Interaktionsmodelle für prototypische und ähnliche Künstler, sowie für beschreibende Eigenschaften.
Basierend auf den Erkenntnissen aus den durchgeführten Experimenten und Evaluierungen wurde die Hauptapplikation dieser Dissertation, nämlich das *Automatically Generated Music Information System* (AGMIS) erarbeitet. AGMIS demonstriert die Anwendbarkeit der entwickelten Techniken auf eine Sammlung von mehr als 600.000 Musikkünstlern indem es eine Web-basierte Benutzerschnittstelle zur Verfügung stellt, mittels derer auf eine Datenbank mit den automatisch extrahierten Informationen zugegriffen werden kann. Wenngleich AGMIS nicht für jeden Künstler und jede Band fehlerfreie Ergebnisse liefert, so weisen die automatischen Methoden der Informationsgewinnung doch einige Vorteile im Vergleich zu jenen auf, die in bereits existierenden Musikinformationssystemen Anwendung finden. Diese basieren nämlich entweder auf arbeitsintensiver Informationsbeschaffung und -verarbeitung durch Experten oder auf dem Wissen von Benutzergruppen, was wiederum eine Verzerrung der Informationen mit sich bringen kann.

Abstract

In the context of this PhD thesis, methods for automatically extracting music-related information from the World Wide Web have been elaborated, implemented, and analyzed. Such information is becoming more and more important in times of digital music distribution via the Internet as users of online music stores nowadays expect to be offered additional music-related information beyond the pure digital music file. Novel techniques have been developed as well as existing ones refined in order to gather information about music artists and bands from the Web. These techniques are related to the research fields of *music information retrieval*, *Web mining*, and *information visualization*. More precisely, on sets of Web pages that are related to a music artist or band, *Web content mining* techniques are applied to address the following categories of information:

- similarities between music artists or bands
- prototypicality of an artist or a band for a genre
- descriptive properties of an artist or a band
- band members and instrumentation
- images of album cover artwork

Different approaches to retrieve the corresponding pieces of information for each of these categories have been elaborated and evaluated thoroughly on a considerable variety of music repositories. The results and main findings of these assessments are reported. Moreover, visualization methods and user interaction models for prototypical and similar artists as well as for descriptive terms have evolved from this work.

Based on the insights gained by the various experiments and evaluations conducted, the core application of this thesis, the *Automatically Generated Music Information System* (AGMIS) was build. AGMIS demonstrates the applicability of the elaborated techniques on a large collection of more than 600,000 artists by providing a Web-based user interface to access a database that has been populated automatically with the extracted information. Although AGMIS does not always give perfectly accurate results, the automatic approaches to information retrieval have some advantages in comparison with those employed in existing music information systems, which are either based on labor-intensive information processing by music experts or on community knowledge that is vulnerable to distortion of information.

Acknowledgments

First, many thanks are due to *Gerhard Widmer* since it was he who gave me the opportunity to work in a very interesting field of research and participate actively in the forming of the *Department of Computational Perception* since its creation in fall of 2004. Furthermore, I want to thank him particularly for supervising and directing my work, for the numerous interesting discussions, for his great flexibility, and for allowing me to participate in many interesting conferences all over the world.
Moreover, I would like to thank *Andreas Rauber* for assessing my work presented in this PhD thesis.
Special thanks are due to all my colleagues, especially *Peter Knees*, *Tim Pohle*, and *Klaus Seyerlehner* for many interesting and fruitful discussions and collaborations. Also *Josef Scharinger*, who always ensured that I went to lunch in time, deserves being mentioned here.
In the context of my PhD studies, I had the chance to meet many interesting people from diverse research communities. Of these I wish to specially thank *Elias Pampalk*, one of my master's thesis supervisors, and *Gijs Geleijnse* who is also doing considerable work in the field of Web mining for music information retrieval.
Furthermore, I would like to express my gratitude to *exalead* for letting me query their search engine automatically. Without their help, especially that of *Julien Carcenac*, I would not have been able to perform the Web crawls required to obtain the large amount of data for this thesis.
Special thanks are due to *Cornelia Schiketanz* for her unconditional support during the time I worked on this thesis and for keeping everything running in Vienna.
Last but not least, I want to thank the students at the *Johannes Kepler University* who participated in one way or another in this thesis, primarily by doing internships.

Contents

1 Introduction and Context **12**

1.1 Problem Description and Motivation for this Thesis . . . 13

1.1.1 Overview of Music Information Systems . . . 13

1.1.2 Motivation for an Automatically Generated MIS . . . 14

1.2 Related Research Areas . . . 17

1.2.1 Music Information Retrieval . . . 17

1.2.2 Web Mining . . . 21

1.2.3 Information Visualization . . . 23

1.3 Contributions . . . 24

1.3.1 Methods . . . 24

1.3.2 Applications . . . 25

1.4 Organization . . . 30

2 Categories of Artist-Related Information **32**

2.1 Relations between Artists . . . 32

2.1.1 Artist Similarity . . . 32

2.1.2 Artist Prototypicality . . . 36

2.2 Descriptive Artist Properties . . . 37

2.2.1 Descriptive Terms . . . 38

2.2.2 Genre and Style . . . 42

2.2.3 Band Members and Instrumentation . . . 44

2.3 Additional Information . . . 44

2.3.1 Song Lyrics . . . 44

2.3.2 Discography and Album Covers . . . 46

3 Techniques Related to Web Content Mining and Information Retrieval **48**

3.1 Web Page Retrieval . . . 48

3.2 Indexing . . . 49

3.3 Information Extraction . . . 52

3.3.1 Co-Occurrence Analysis . . . 53

3.3.2 Backlink/Forward Link Analysis . . . 54

3.3.3 Term Weighting . . . 57
3.3.4 Natural Language Processing for Determining Band Members . . . 58
3.3.5 Image Retrieval . . . 61

4 Techniques Related to Information Visualization 64
4.1 Visualization Methods for Relations Between Artists . . . 64
4.1.1 Self-Organizing Maps . . . 65
4.1.2 Multi-Dimensional Scaling . . . 66
4.1.3 Similarity Networks . . . 67
4.1.4 Continuous Similarity Ring . . . 69
4.2 Visualization Methods for Hierarchical Data . . . 72
4.2.1 Treemap . . . 72
4.2.2 Hyperbolic Browser / Hyperbolic Tree . . . 72
4.2.3 Circled Fans . . . 74
4.2.4 Sunburst . . . 76
4.2.5 Stacked Three-Dimensional Sunbursts . . . 78
4.2.6 Co-Occurrence Browser . . . 80

5 Experiments and Evaluation 86
5.1 Test Collections . . . 86
5.2 Query Schemes . . . 87
5.3 Experiments . . . 88
5.3.1 Artist Similarity . . . 88
5.3.2 Artist Prototypicality . . . 101
5.3.3 Descriptive Artist Properties . . . 105
5.3.4 Band Members and Instrumentation . . . 110
5.3.5 Album Covers . . . 115
5.3.6 Visualizing Artist-Related Web Pages with the COB . . . 118

6 AGMIS: An Automatically Generated Music Information System Based on Information from the Web 122
6.1 Artist Collection . . . 122
6.2 Data Acquisition . . . 123
6.2.1 Querying . . . 123
6.2.2 Fetching . . . 124

6.2.3 Indexing 124
6.3 Database Design 127
6.4 Information Extraction 127
6.5 Web User Interface 129
6.6 Computational Complexity 132

7 Conclusions **133**

8 Outlook and Future Work **135**

List of Figures

1.1 User interface to conveniently access most of the functionality provided by the *CoMIRVA* framework. . . . 26
1.2 User interface of the *Co-Occurrence Browser*. . . . 27
1.3 Web-based user interface provided by *AGMIS*. . . . 28
1.4 *nepTune* user interface in the mode where artist and track names are displayed. . . . 29
1.5 *Traveller's Sound Player* extended with the visualization of genre distributions. . . . 30

4.1 Music Browser visualization using Multi-Dimensional Scaling to organize songs. . . . 68
4.2 Two Probabilistic Network visualizations based on Multi-Dimensional Scaling and random graphs. . . . 69
4.3 Continuous Similarity Ring visualization based on prototypical artists for 14 genres. . . . 71
4.4 Treemap visualization of a file system. . . . 73
4.5 Hyperbolic Tree visualizations of similarity relations for the bands *Die Ärzte* and *Black Sabbath* as extracted from last.fm. . . . 74
4.6 Circled Fans visualization of similarity relations for the band *Die Fantastischen Vier* derived from band name co-occurrences on Web pages. . . . 76
4.7 Sunburst visualization of frequently co-occurring terms on Web pages about the band *Iron Maiden*. . . . 77
4.8 Stacked Three-Dimensional Sunbursts visualization using three layers. Color is used to emphasize the data dimension encoded in the arcs' angular extent. . . . 84
4.9 Stacked Three-Dimensional Sunbursts visualization using three layers. Color is used to emphasize the data dimensions encoded in the arcs' heights. . . . 84
4.10 COB used to illustrate 198 Web pages of the band *Iron Maiden*, which are clustered according to co-occurring terms. . . . 85

5.1 Accuracies, in percent, for single and compound similarity measures using 9-NN t_{15} validation and the confidence filter on *C224a14g*. . . . 96
5.2 Accuracy for the single AIT measure, plotted against percentage of classified artists for different training set sizes and 9-NN classification on *C224a14g*. . . . 96
5.3 Accuracies for 9-NN classification experiments, in percent, for different combinations of the query schemes AIT, M, and MR and different training set sizes on *C224a14g*. . . . 97

5.4 Confusion matrix for the co-occurrence approach using 3-NN classification and the compound AIT+MGS measure on *C103a22g*. 99
5.5 Confusion matrix for the TF·IDF approach using 3-NN classification and the MGS scheme on *C103a22g*. 100
5.6 Confusion matrices of the classification task for each of the three prototypicality ranking approaches. 103
5.7 Confusion matrices of the classification task using the BL/FL approach with penalization of exorbitant popularity, shown for every genre. 103
5.8 Precision/recall-plot for the band members and instrumentation detection approach on collection *C51a240m*, using exact string matching. 113
5.9 Precision/recall-plot for the band members and instrumentation detection approach on collection *C51a499m*, using exact string matching. 113

6.1 Data processing diagram of AGMIS. 123
6.2 Entity relationship diagram of the AGMIS database. 128
6.3 List of artists provided by AGMIS for the search term "Metal". 130
6.4 Part of the user interface provided by AGMIS for the artist *B.B. King*. 130
6.5 Complete artist information page returned by AGMIS for *B.B. King*. 131

List of Tables

3.1 Synonyms for instruments and roles used in band member detection. 59

5.1 Query schemes used in the experiments conducted. 88
5.2 Evaluation results of intragenre/intergenre similarities using co-occurrence analysis based on page counts. 91
5.3 Evaluation results of intragenre/intergenre similarities using co-occurrence analysis as proposed in [Zadel and Fujinaga, 2004]. 91
5.4 Evaluation results of intragenre/intergenre similarities using co-occurrence analysis based on retrieved page content. 92
5.5 Evaluation results of intragenre/intergenre similarities using TF·IDF vectors. 92
5.6 Accuracies, in percent, for different training set sizes using the TF·IDF approach on collection *C224a14g*. 95
5.7 Accuracies, in percent, for k-NN evaluations using co-occurrence analysis on collection *C103a22g*. 98
5.8 Accuracies, in percent, for k-NN evaluations using the TF·IDF approach on collection *C103ag22*. 99
5.9 The 10 top-ranked artists of the genres "Heavy Metal" and "Folk" for each of the three prototypicality models. 104
5.10 Overall genre-specific accuracies for the three prototypicality models. 104
5.11 Spearman's rank-order correlations between the ground truth ranking by AMG and the rankings obtained with the prototypicality ranking approaches. 106
5.12 Results of the user study on different term weighting functions. 107
5.13 Continuation of Table 5.12. 108
5.14 Overall precision and recall of the predicted (member, instrument)-pairs, in percent, for different query schemes and string distance functions on collections *C51a240m* and *C51a499m*. 112
5.15 Upper limits for the recall achievable on collections *C51a240m* and *C51a499m*. 112
5.16 Evaluation results for album cover detection approaches on collection *C225b*. 117
5.17 Evaluation results for album cover detection approaches on collection *C3311b*. 118
5.18 For each participant, the time needed to finish each task of the COB user study, measured in seconds. 120

6.1 List of genres used in AGMIS with the corresponding number of artists and their share in the complete collection, in percent. . . . 125
6.2 Amount of artists for which no Web pages were found (Zero-Page-Count-Artists). . . 125
6.3 Retrieved Web pages and empty Web pages. . . . 126
6.4 Median and mean of available Web pages (according to page counts) and mean of actually retrieved Web pages. . . . 126
6.5 Some running times of tasks performed for the creation of AGMIS. . . . 132

A-1 Composition of collection *C224a14g*. . . . 153
A-2 Continuation of Table A-1. . . . 154
A-3 Composition of collection *C112a14g*. . . . 155
A-4 Composition of collection *C103a22g*. . . . 156
A-5 Continuation of Table A-4. . . . 157
A-6 Composition of collection *C103a13g*. . . . 158
A-7 Continuation of Table A-6. . . . 159
A-8 Distribution of genres and tiers given by AMG in collection *C1995a9g*. . . . 159
A-9 List of artists in collections *C51a240m* and *C51a499m*. . . . 159
A-10 Artist and album names in collection *C255b*. . . . 160
A-11 Continuation of Table A-10. . . . 161
A-12 Continuation of Table A-11. . . . 162
A-13 Continuation of Table A-12. . . . 163
A-14 Continuation of Table A-13. . . . 164
B-1 Accuracies, in percent, for k-NN evaluations using the TF·IDF approach on the AGMIS collection. . . . 166

Chapter 1

Introduction and Context

Over the past few years, the research field of *music information retrieval* (MIR) has encountered a remarkable gain in attention. This can be partly ascribed to technological advances, e.g., portable music players with huge hard disks have become available, which raised the question how collections of thousands of music pieces can be organized and accessed efficiently for the highest benefit of the user. Furthermore, the remarkable attention the music industry has been given in the media, e.g., the discussion about digital rights management or various law suits in the context of illegal file sharing in peer-to-peer networks, is playing an important role in making music information retrieval popular since the mentioned topics are strongly related to this field of research.

One key task in MIR research is the automatic extraction of information about pieces of music or music artists. The traditional approaches to this problem rely on the analysis of the audio signal to derive various features describing the sound of a piece of music. In cases where no audio files are available, however, such an audio analysis is obviously not feasible. An alternative way to obtain music-related information is based on the extraction of so-called *cultural features*. MIR research that is based on such cultural features is a relatively young subfield of MIR since first attempts in this direction have not been made until the early 2000s, cf. [Cohen and Fan, 2000], [Ellis et al., 2002], [Whitman and Lawrence, 2002], [Whitman and Smaragdis, 2002]. Although only fuzzily defined in the literature, the term "cultural features" usually denotes features which reflect the knowledge or opinions of a large number of people. Frequently these are members of a common community, in which case such features are also called "community metadata". This kind of data can be derived from a wide variety of sources. For example, lists of purchased music from (online) music stores[1], music collections that are made available via music sharing services, cf. [Ellis et al., 2002], [Whitman and Lawrence, 2002], and playlists of radio stations and compilation CDs, cf. [Pachet et al., 2001], represent some sources that were frequently used in the early days of cultural feature extraction for MIR.

In the past few years, much work has been devoted to exploit the vast resources offered by the World Wide Web (WWW). It is generally assumed that the information provided by the zillions of Web pages and services of the WWW also represents a kind of cultural knowledge and, therefore, can

[1] Such lists are often used for collaborative filtering purposes in music recommender systems to offer the user information of the form "Customers who bought this item also bought the following".

be beneficially used to derive cultural features for MIR tasks. The traditional source for Web-based feature extraction are HTML pages gained by crawling the Web. This source was already exploited for MIR in [Cohen and Fan, 2000], where Web pages of the user's favorite artists were downloaded and analyzed. The recent developments and applications of novel Web technologies enabled using other, more tailored, sources for data acquisition in Web-based MIR, like Web logs, cf. [Celma et al., 2006], RSS feeds, cf. [Celma et al., 2005], or Web sites like *Epinions.com* [epi, 2007] that are specialized on music reviews, cf. [Hu et al., 2005], [Hu et al., 2006]. However, as the contribution of HTML pages to the total amount of information offered by the WWW is still higher than that of specialized Web services and applications, the techniques elaborated in the context of this thesis focus on gathering useful information from static as well as from dynamically generated Web pages.

1.1 Problem Description and Motivation for this Thesis

The principal aims of this PhD thesis are automatically finding and extracting data about music artists from the Web, the processing of this data in order to derive useful information, and the representation of this information in the form of a *music information system (MIS)* that is automatically generated. The derived information will be of very different kinds. To give some examples, this thesis will present – among other things – methods for harvesting information about the similarity of two artists, for gathering terms that describe the style of the music an artist performs, for estimating the importance of an artist for a specific genre, for deriving information about band members and instrumentation, and for finding images of album covers.

1.1.1 Overview of Music Information Systems

Music information systems are – usually Web-based – services that offer various kinds of music-related information, commonly on different levels. Such levels are typically artist, album, or track names. The information which is offered to the user obviously varies according to the implemented functionalities of the MIS, but also according to these different levels. For example, typical information offered on the artist/band level includes biographies, band pictures, tour dates, band members, instruments, and country of origin. On the album level, images of the album artwork are usually provided, as well as release dates, album reviews, and links to online stores. Information on the track level typically includes lyrics, similar tracks, and sometimes a preview snippet of the corresponding digital audio file (often in a low-quality version). Similarity information is also frequently provided at the artist level. Furthermore, user-assigned tags that describe artists or music, or relations between the user and the music or artist are often present on all of these levels.

The existing music information systems can be broadly divided into two categories according to the way their databases are populated and maintained. These tasks can either be performed by *experts* or by a *community*. Probably the most prominent example of an expert-based system is the *All Media Guide* [amg, 2007a] (originally known as *All Music Guide*, or short, AMG), which was founded in 1991 "to help consumers navigate the increasingly complex world of recorded music and discover the very best recordings" [amg, 2007b]. Since 1995, the AMG is accessible via a Web site. Other popular examples for expert-based systems are those offered by Web search engines, namely *Yahoo! Music* [yah, 2007b] and *MSN Music* [msn, 2007]. It seems, however, that a large part of the content provided by theses two systems originates from the AMG. As for the endeavors of the major search engine *Google*, in 2006, they launched a music search service, but removed it from their official *Google Products* page [goop, 2007] after only some weeks. At the time when this thesis was written, the only thing that remained of Google's music service was a hidden servlet providing search for and display of artists, albums, and songs.

With the advent of the so-called "Web 2.0", not only has it nearly become a must for the providers of existing systems to incorporate some sort of a user participation model into their services[2], but the movement towards the "Web 2.0" also led to the emergence of new music information systems, which entirely rely on the power of the crowd. This means that the provided information solely (or at least largely) depends on the participation of the users. The most notable MIS of this type are *last.fm* [las, 2007] and *Discogs* [dis, 2007]. Founded in 2002, last.fm is one of the most comprehensive MIS these days and provides a couple of additional features beyond the pure music information capabilities mentioned at the beginning of this section. For example, the user can listen to personalized radio stations that are created from the user's profile, which in turn is constructed using the so-called "scrobbling", a procedure that refers to the automatic transmission of information on the user behavior in the context of his or her music player. Discogs, launched in 2000, offers similar information, albeit has a stronger focus on providing links to online music stores.

1.1.2 Motivation for an Automatically Generated MIS

However, both of the aforementioned categories of MIS – expert-based as well as community-based – have drawbacks. The most severe one is the tremendous amount of labor which has to be invested in order to maintain the databases of existing MIS. In fact, regardless of the people who perform these maintenance operations – be it experts as in the case of the All Music Guide or a community of interested people as in the case of *last.fm* – keeping the probably very large database of an MIS up-to-date

[2] At its most primitive form, such a participation model can be a user rating of tracks or albums. Such rating information can then be used to recommend new music according to the user's taste.

is an extremely labor-intensive task.[3] This becomes especially apparent when taking into account the highly dynamic nature of the domain of music information, with its frequent album releases, creations, dissolvings, and reunions of bands, temporary collaborations of artists in different bands, and many other events that require updating the database.

Expert-based MIS further often suffer from another shortcoming, which is their *cultural bias*. Considering the most prominent example, the AMG, once again, this bias towards the cultural context and subjective opinions[4] of the music experts that maintain the MIS is illustrated well by the fact that, for a long time, no information about some very popular, albeit non-American, bands could be found in the AMG. Indeed, bands which are only known in regions outside the United States of America, were not covered by the AMG, regardless of their popularity. For example, although very popular for their style of music for nearly 10 years, the German medieval folk-rock band *Schandmaul* was not known to the AMG until early 2007. The same is true for many other bands which do not fit into the commonly (mis-)used pop/rock genre, but nevertheless enjoy high regional popularities. AMG fortunately became aware of this problem, and therefore, since 2006, intends to broaden its coverage to music from all regions of the world. However, a quick, non-representative look at some of the author's favorite bands revealed that these intentions have only been moderately successful as there is still a lot of information missing. Moreover, the author discovered many flaws in the information provided for artists like *Schandmaul*, *Faun*, or *Cultus Ferox*: misspellings of track and album names, wrong album release years, wrong periods of activity, inconsistent label names – just to mention some.

On the other hand, MIS whose data is provided by a community of people interested in music usually do not suffer from a lack of information, provided that the community is large and covers a broad variety of different music tastes. However, such community-based MIS, like last.fm, also face some serious problems. The most problematic ones are the *cold start problem* and *social tagging vandalism*. As for the former, the cold start problem refers to the fact that newly added entities (e.g., new artists, albums, or tracks, but also users which are new to the system) usually show a lack of associated information (e.g., user tags or data about users' playing behavior concerning a specific track). Due to this data sparseness, important functionalities like recommending similar artists or presenting descriptive terms are either not available or work very poorly during this cold start phase. However, as the amount of information available about the added entitiy grows, so usually does their accuracy, and also the accuracy of derived information, like similarities between artists or between songs. Obviously, the preconditions for overcoming this cold start phase are a large and diversified community supporting the MIS and a

[3] At the time when these lines were written (2007-12-05), for example, AMG's coverage statistics reported 1,053,328 artists, 1,345,828 albums, and 11,204,711 individual tracks.

[4] Such subjective opinions obviously are also influenced by the environment, thus by the cultural context, of the experts.

certain popularity or activity of the new entity.

Another threat for information systems that rely on the knowledge of the crowd, and therefore also for community-based MIS, is vandalism. An example for this social tagging vandalism can be found when looking at the top-artists tagged with "brutal death metal" on last.fm. While the author is writing these lines, the most important representative for this extreme form of metal music according to last.fm is *Paris Hilton*, which is definitely not true. It seems that a bunch of people disliking Paris Hilton has injected erroneous information about this artist.

As already mentioned, the quality of the content provided by a community-based MIS strongly depends on the diversity of interests of the participating users. If a large number of (active) participants tend to listen to only a very restricted style of music or set of artists, this commonly has a bad impact on the overall quality of the system since it overemphasizes certain groups of artists. This problem is known as *population bias*. last.fm, for example, has a strong focus on music from the genre metal, which becomes especially evident when looking at the top-played artists and the most frequently applied genre tags. The main issue of this population bias is that it leads to uneven granularities of descriptions (annotations via tags) for different groups of artists, and may further distort the results of music recommendation algorithms.

The aforementioned problems arising in the context of systems that involve user behavior monitoring and analysis are not the only ones. More issues of such systems, especially of music recommender systems, can be found in [Celma and Lamere, 2007].

To summarize, both categories of MIS suffer from more or less severe drawbacks. First, both are not robust against biased information, regardless of whether this bias is caused by the cultural context of a group of experts or by the overemphasis of certain music styles by a community. Even though expert-based MIS are commonly said to offer more reliable information, they often lack a broad coverage that spans all musically relevant geographical regions and music styles.[5] On the other hand, community-based systems have to face the problems of hacking and vandalism as well as a lack of information on newly added items.

The driving force behind this thesis was the author's motivation to build a music information system that does neither rely on music experts nor on a specialized community. Instead, one of the world's largest and most diversified communities, that formed by the information publishers on the World Wide Web is considered. To this end, the MIS developed in the context of this thesis makes use of the information provided on the zillions of Web pages available on the WWW.

The principal choice of building an MIS by harvesting the WWW instead of relying on expert or com-

[5] Similar observations on the accuracy of information can be made when comparing "classical" encyclopedias with the community-based *Wikipedia* [wik, 2007b], an online encyclopedia where anyone can edit articles.

munity knowledge entails several advantages, but also some challenges, which have to be faced. The most obvious pro of this choice is that the Web contains information about nearly everything. Thus, in contrast to expert-based systems, an MIS that relies on Web-based MIR techniques should not be confronted with the problem of lacking information. On the other hand, the data available on the Web is extremely noisy and often reveals ambiguous information. Hence, determining which Web pages are eligible and reliable information sources as well as extracting the right information from these sources are challenging tasks. As for the problem of the cultural bias, compared to systems which are maintained by experts sharing the same or a similar cultural background, an MIS built on Web data alleviates this cultural bias. Even though it is certain that some regions of the world are still underrepresented on the Web, its usage, in particular for information publishing purposes, is registering an enormous boost, especially in the yet underrepresented areas of the world, like Asia. The automatically generated music information system proposed here is further more robust against hacking and vandalism than a community-based approach since hacking individual Web pages and deliberately interspersing spurious information usually do not influence the information extraction process more than does accidentally published wrong information. Of course, this issue is also a matter of scale. If, for a particular artist, 20 relevant Web pages are found, hacking of one or a few pages poses a more severe problem than for an artist with millions of Web pages available. On the other hand, it is unlikely that Web pages about such an unknown artist will be the target of a hacking attack. Above all, the most decisive advantage of a system that automatically harvests music-related information from the Web is its self-maintenance, and hence, the overcoming of the need for people that perform the labor-intensive task of manually feeding information into the system – be it either experts or music fans.

1.2 Related Research Areas

The techniques elaborated and applied in the context of this thesis are strongly related to the fields of music information retrieval, Web mining, and information visualization. In the following, a brief overview of these research areas is given in order to unveil the context of this work.

1.2.1 Music Information Retrieval

From a most general point of view, music information retrieval (MIR) is concerned with the extraction, analysis, and usage of information about any kind of music entity (for example, a song or a music artist) on any representation level (for example, audio signal, symbolic MIDI representation of a piece of music, or name of a music artist). The following list is the author's attempt to group the heterogeneous research areas within the field of MIR. However, the author does not claim to exhaustively cover all

areas, rather to point out some important subfields of MIR.

Feature Extraction

Extraction of meaningful features from the audio signal or from other information sources related to music, like specific Web pages or metadata provided by music publishers. Audio signal-based features are frequently categorized into low-level and high-level features.[6] Examples of the former ones are energy in different frequency bands, zero crossing rate of the waveform, spectral centroid, or spectral flux. High-level features, in contrast, describe properties like rhythm, tempo, melody, harmony, or tonality. In general, low-level features are closely related to the audio signal, whereas high-level features represent more abstract properties of music. A more detailed discussion of audio features can be found in [Pohle, 2005]. In contrast, features derived from metadata are, for example, counts of user ratings or tags as well as frequencies of terms appearing on artist-related Web pages.

Similarity Measurement

Defining and calculating similarities between a certain category of music entity also plays a vital role in MIR. In order to yield appropriate results, the measure must be applied to meaningful features. On the other hand, even if some sort of expedient feature is at hand, not any similarity measure may give worthwhile results. Hence, the choice of a good combination of feature data and similarity measure is crucial when designing MIR applications. Such applications of similarity measurement include *automatic music playlist generation* or *music recommender systems*.

Audio Fingerprinting

This topic is to some extent related to feature extraction since its main challenge is to construct a unique and robust identifier of a piece of music, that is usually created from the piece's audio signal. Given only a short (and often noisy) audio excerpt from a piece of music, a good fingerprinting system is capable of identifying the original piece correctly. Also *copyright infringement detection* makes use of fingerprinting as this task requires finding differently encoded versions of the same piece of music.

Voice and Instrument Recognition

This subfield's main concern is the extraction and analysis of typical acoustic patterns that can be used to discern different music instruments and voice, given only the waveform of a piece of music or any other audio stream. Once such patterns are identified, they can be used to categorize unknown

[6] In this PhD thesis, the terms "signal-based", "audio-based", and "audio signal-based" will be used interchangeably.

music material according to its instrumentation. These tasks are obviously related to feature extraction and classification. A comprehensively discussed special case is the automatic distinction between music and speech, which is often called *speech/music discrimination*, e.g., [Scheirer and Slaney, 1997], [Seyerlehner et al., 2007].

Structural Analysis, Alignment, and Transcription

These research areas are mainly concerned with the acoustic analysis of pieces of music with the objective of revealing their structure, at different granularity levels. Structural analysis involves feature extraction in a first stage. Subsequently, *segmentation* of the audio stream is performed, i.e., boundaries for partitioning the piece of music under consideration are fixed. Based on the results of the segmentation process, recurring acoustic patterns are sought. This is commonly accomplished by calculating *self-similarities*, i.e., similarities between the different segments of one and the same piece of music. Such self-similarities serve as indication for certain elements of a piece of music (e.g., verse, chorus, or bridge). Applications of structural analysis include *music summarization*, generation of *structural descriptors* that facilitate the annotation of music pieces, and *audio synthesis*. Furthermore, the techniques used in structural analysis considerably contribute to solving the problem of *audio and lyrics alignment*. A much more detailed elaboration on the topic can be found in [Ong, 2005].
Related to structural analysis, albeit considering a finer-grained level, is the topic of *audio to score alignment*, which is also known as *score following* if performed in real-time. Also the alignment or matching of different interpretations of the same piece of music falls in this category. These are quite challenging tasks as different interpreters often vary heavily in their styles and performances. Another related task is that of *audio to score transcription*, which aims at recognizing individual notes or chords in an audio waveform. Even for monophonic music this is by no means a trivial task. But it gets even harder when polyphonic material, like orchestra performances, is considered.

Music Tonality Analysis

This subfield of MIR is concerned with the extraction of tonal information from music. Important aspects in this context are again feature extraction and *key finding*. Related work can be found, for example, in [Gómez, 2006].

Optical Music Recognition

Abbreviated OMR, this research area aims at converting images of scanned sheet music into a musically meaningful digital format, like a representation as a MIDI file. The main challenges that have to

be faced are recognizing and interpreting the music symbols on sheets of music. Poor quality of (historic) sheets as well as image distortions arising from the scanning process aggravate this task considerably. More comprehensive elaborations can be found, for example, in [Bainbridge, 1997] and [Byrd and Schindele, 2006].

Classification and Evaluation

An important task is the evaluation of approaches proposed for the aforementioned problems in MIR. To this end, a predefined *ground truth*[7] is commonly used, against which the approach under consideration is evaluated. To perform evaluation, a classification setting is often constructed, and classification experiments are conducted accordingly. Consider, for example, a new algorithm (that makes use of feature extraction and similarity measurement techniques) to automatically categorize pieces of music according to genre, mood, or instrumentation is proposed. Evaluation, in this case, would probably be performed by comparing the automatically predicted categories for the pieces of music in the dataset with the categories given by the ground truth. The performance of the proposed algorithm is then estimated by calculating some kind of *quality measure* on these comparisons. As classification and evaluation are fundamental topics in *machine learning* and fill whole books, they cannot be covered in more detail in this introductory section. More information on these topics can be found, for example, in [Bishop, 2006]. However, the strong dependence on such topics reflects the multidisciplinarity of MIR.

User Interfaces, Visualization, and Interaction

Also encompassed by MIR is the design and development of intelligent user interfaces to access music collections. The principal aim of this subfield is to provide intuitive means of navigating in music repositories, beyond the standard approach followed by today's music players, which is text-based browsing through an artist–album–track or a genre–artist–album–track hierarchy. Creating such alternative user interfaces is a multidisciplinary challenge as it usually involves applying *information visualization* techniques to visualize music-related information gained by applying other MIR techniques, e.g., music descriptors or automatically generated playlists. Furthermore, issues of *human-computer interaction* have to be considered in this context, as well as *usability* questions.

[7] In the context of MIR, the term *ground truth* usually refers to a dataset comprising the music entities (or an identifier of these entities) and an assignment of suitable properties to these entities (according to the problem which is addressed by the approach to be evaluated). Such a ground truth should obviously be as objective as possible, although, in practice, this is a hardly achievable goal. Examples are a set of songs annotated with mood descriptions or a mapping between artist names and the predominant genre an artist performs.

1.2.2 Web Mining

The remarkable increase of the importance of the World Wide Web in our daily lives and the associated progression of available Web pages have led to the strange fact that nowadays, even though the WWW provides an incredible amount of information, it is often quite difficult for the user to find the desired one. Thus, *Web mining* techniques that, broadly speaking, aim at extracting useful information from the WWW, have become more and more important over the last few years. Web mining is related to *data mining* and *text mining*, but also differs considerably. A good description of the similarities and differences can be found in [Liu, 2007].

> "It [Web mining, note from the author] is related to data mining because many data mining techniques can be applied in Web content mining. It is related to text mining because much of the Web contents are texts. However, it is also quite different from data mining because Web data are mainly semi-structured and/or unstructured, while data mining deals primarily with structured data." [web, 2007]

In short, the principal difference between these related research areas is the type of the input data. While data mining techniques are tailored to yield good results on well-structured and organized data, e.g., on data given in the form of relational databases, text mining approaches have to deal with the unstructured nature of texts. At this point, the reader may argue that natural language texts usually should follow the rules of grammar. However, compared to the stringent organization of data given by a relational database model, texts can be considered poorly structured. Moreover, natural language texts, unlike relational databases, offer no easily discernible semantics – at least from the point of view of a computer. Compared to these two diametrically opposed forms of input data structuredness, Web mining lies somewhere in between. Indeed, Web pages are often said to have a semi-structured nature. This is due to the fact that they largely consists of unstructured text on the one hand. On the other hand, however, Web pages show at least some primitive form of structure as they use tags to organize the provided information. Moreover, hyperlinks are used to organize a set of Web pages among each other.

The research field of Web mining is commonly divided into the three subfields *Web content mining*, *Web structure mining*, and *Web usage mining*, which are briefly introduced in the following. A comprehensive elaboration on the topic of Web mining can be found in [Chakrabarti, 2002] or, more recently, in [Liu, 2007].

Web Content Mining

Extracting useful information and discovering knowledge from the content of Web pages are the main goals of this field. As Web page content usually consists of multimodal data (e.g., text, image, audio, and video), Web content mining involves applying techniques from the wide field of *information retrieval*, which are tailored to deal with the various modalities of information sources. Taking into account the fact that the predominant source of information on Web pages is still texts, techniques related to *text mining* certainly play the most fundamental role in this context. Such techniques include, for example, approaches to *text categorization*, *text clustering*, and *text summarization*. Each of them can be applied to the textual content of Web pages. Furthermore, *natural language processing* (NLP) is a related field as it tries to capture the semantics of natural language texts from the point of view of a computer.

Web Structure Mining

This subfield focuses on exploiting the hyperlink structure of the Web. As this structure can be regarded as a graph whose nodes are the Web pages and whose edges are the links between them, techniques from *graph theory* can be applied to analyze this structure. A more detailed overview of this topic can be found in [Fürnkranz, 2002]. By analyzing the links in a set of Web pages, the importance of each page can be estimated, which evidently is a crucial prerequisite for *building search engines*. The most prominent example in this context is probably the *PageRank* algorithm used by Google's search engine, cf. [Page et al., 1998]. Apart from its use in search engines, other application areas of Web structure mining include *Web page clustering*[8] and *Web page categorization*.

Web Usage Mining

The focus of this research area is on discovering patterns in data about the user behavior in the context of Web access. More precisely, the users' interactions with Web resources are monitored and analyzed. The main source of such interaction data is server access logs, which store every HTTP request issued by the client. From such data, a wide variety of information can be derived, e.g., amount of time spent on specific pages or sections of a Web site, typical date and time of a certain category of user activity, access frequency of a Web page, or information on paths through a Web site typically taken by the user. Such information plays a vital role for applications of Web usage mining, like *Web personalization*, i.e., providing personalized Web content to the user, and the *analysis of customer and visitor behavior* in e-commerce systems. Probably the most well-known task in the context of personalization is *col-*

[8] The discovered clusters can be interpreted as communities.

laborative filtering, which aims at making recommendations to the current user, based on similarities between his/her behavior profile (that, among other things, records the user's purchases and estimates his/her interests) and that stored for other users. More details on Web usage mining can be found in [Srivastava et al., 2000] and [Mobasher, 2004].

The techniques elaborated and applied in the context of this thesis can be mainly classified as Web content mining focused on MIR, especially the *co-occurrence analysis* of terms on Web pages for deriving artist similarity and the determination of band members using an NLP approach.

1.2.3 Information Visualization

Information visualization (InfoVis) is an interdisciplinary research area that draws upon *computer graphics*, *user interface design* and *human-computer interaction*. The primary concern of InfoVis is the visual display of data, whose dimensionality and/or quantity is commonly high. Thus, the crucial property of a good information visualization is that it offers the user a better understanding of the data under consideration as he/she is offered by a purely textual data representation.[9] Choosing a suitable visualization method for a given data set is by no means a trivial task, although there exist design guidelines and principles as well as standard visualization approaches for certain categories of data structures.
As for related literature, an introduction to information visualization on a basic to intermediate level can be found in [Spence, 2007]. [Ware, 2004] covers the topic from perceptual and cognitive points of view, whereas [Tufte, 2001] focuses on giving design guidelines for visualizing data by means of discussing examples of good and bad visualizations, according to the author. In contrast, [Harris, 1999] offers an encyclopedic collection of information illustration methods, and is thus suited to get a very quick overview of several, mostly quite simply, visualization techniques.

In the context of this PhD thesis, the data to be visualized mainly consists of information on term co-occurrences, more precisely, sets of terms that are assigned to a number of Web pages. Since such data can be interpreted as a hierarchical structuration of sets of Web pages, the author will focus on visualization techniques for hierarchically organized data in the respective Chapter 4. In order to illustrate the co-occurrence data, a novel visualization technique, namely the *Stacked Three-Dimensional Sunbursts*, has been developed.

[9] Note, however, that sometimes a textual representation may be the best visualization method, i.e., best suited for the representation of the data under consideration.

1.3 Contributions

Over the past three and a half years, one of the author's principal intention has been bringing together the fields of Web content mining and music information retrieval. Looking at the contributions of probably the most important forum for MIR, the annual "International Conference on Music Information Retrieval", cf. [ism, 2007], it becomes apparent that the possibilities of using the Web as a data source for MIR have been neglected for a long time.

In fact, except for some exceptional publications, it was not before the mid 2000s that first steps to considering Web-based MIR were taken by the MIR community. Lately, fortunately, more and more MIR researchers tend to consider MIR-related information gathering from the WWW for their scientific work. The rising popularity of "Web 2.0" applications further reinforced this trend, and at ISMIR 2007, a quite considerable number of contributions dealt with the use of the Web for MIR tasks.

The author of this thesis is one of several researchers who have been actively contributing to bringing together Web retrieval and music information retrieval and in establishing music-related Web information retrieval as part of MIR. Some results of these endeavors are presented in this PhD thesis. More precisely, this thesis contributes to the state-of-the-art in Web-based MIR research in various ways. First, by developing Web-based information retrieval and visualization approaches that are tailored to deal with the domain of music artists. Second, by thoroughly evaluating these approaches on various data sets of different difficulties. Third, by creating a number of applications that make use of these approaches, especially by integrating the results of applying the approaches to a large set of music artists into a system that offers automatically extracted information via a Web application.

The contributions can be divided into *methodological contributions* and *applications* developed in the context of this thesis, which is elaborated in a more precise manner in the following.

1.3.1 Methods

A quick overview of the main methodological contributions of this thesis is given in [Schedl et al., 2008]. An even briefer summarization of these methods is given in the following. All of the elaborated methods are mainly the author's work, although most of them benefited from fruitful discussions with colleagues.

Co-Occurrence Analysis

Term co-occurrences on artist-related Web pages are used to estimate the similarity between the corresponding artists.

Artist Prototypicality Measurement

The concept of *prototypicality* of an artist for a certain genre is defined, and information on co-occurrences of artist names on Web pages are used to derive methods for measuring this prototypicality.

Band Member and Instrumentation Detection

Primarily based on *named entity detection* and *natural language processing*, a method that analyzes band-related Web pages in order to determine the members of a music band and the instruments they play is elaborated.

Album Cover Retrieval

Context-based as well as content-based methods to automatically gather album cover artwork from related Web pages are developed.[10] Moreover, a combination of content-based filtering and context-based selection is proposed.

Stacked Three-Dimensional Sunbursts

The visualization method of *Stacked Three-Dimensional Sunbursts* is elaborated. It provides a means of illustrating the hierarchical structure imposed on a set of (Web) documents by recursively clustering this set according to term co-occurrences.

1.3.2 Applications

The applications developed in the larger context of this PhD thesis can be categorized into those developed solely or largely by the author and those for which the author elaborated and implemented extensions or otherwise participated considerably in the development process.

CoMIRVA

Into the former category falls the open source framework and toolkit *Collection of Music Information Retrieval and Visualization Applications* (CoMIRVA), presented in [Schedl et al., 2007a] and downloadable from [com, 2007]. CoMIRVA provides an environment for performing MIR tasks and offers various implementations of algorithms, ranging from feature extraction (audio signal-based as well as Web-based) and similarity calculation to data projection and visualization methods. Most of the approaches presented in this PhD thesis also found their way into the CoMIRVA framework. Although

[10] "Context-based" here refers to the context given by the surrounding area of the HTML tag via which an image is embedded in a Web page, whereas "content-based" refers to the perceivable content of the image.

mainly developed and maintained by the author, many students contributed to the CoMIRVA project. A special contribution has been made by *Klaus Seyerlehner*, who implemented two state-of-the-art high-level audio feature extractors. A screenshot of CoMIRVA's user interface is given in Figure 1.1. The right part of the UI is occupied by lists of data matrices and metadata vectors, whereas the large area to the left of these lists serves as visualization area. In this screenshot, the visualization area depicts a Sunburst illustration, cf. Subsection 4.2.4.

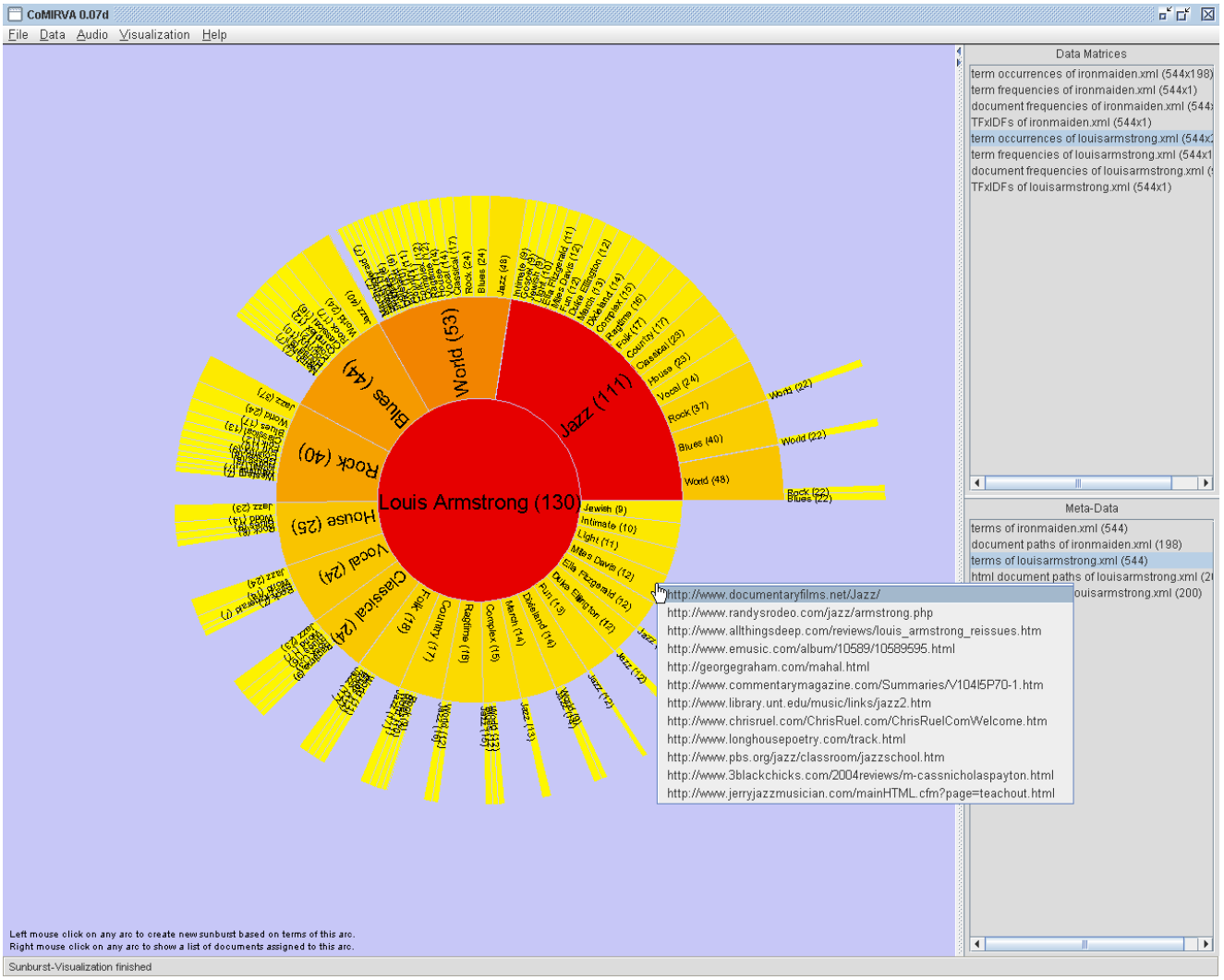


Figure 1.1: User interface to conveniently access most of the functionality provided by the *CoMIRVA* framework.

COB

The *Co-Occurrence Browser* (COB) is a sole development of the author and provides a user interface to organize and browse a set of Web pages related to a certain topic, which will usually be a music artist

or band in the context of this thesis. COB uses the visualization method of the formerly introduced Stacked Three-Dimensional Sunbursts to structure the input set of Web pages according to terms which have been attributed a high importance by some measure. Furthermore, by extracting information about the multimedia content found on the processed Web pages, COB is able to illustrate the amount of certain categories of multimedia files and organize the Web pages accordingly. A more comprehensive introduction to COB can be found in [Schedl et al., 2007b] and in Sections 4.2.5 and 4.2.6 of this thesis. To get a first impression, Figure 1.2 depicts a screenshot of the COB generated from a set of Web pages about the band *Bad Religion*.

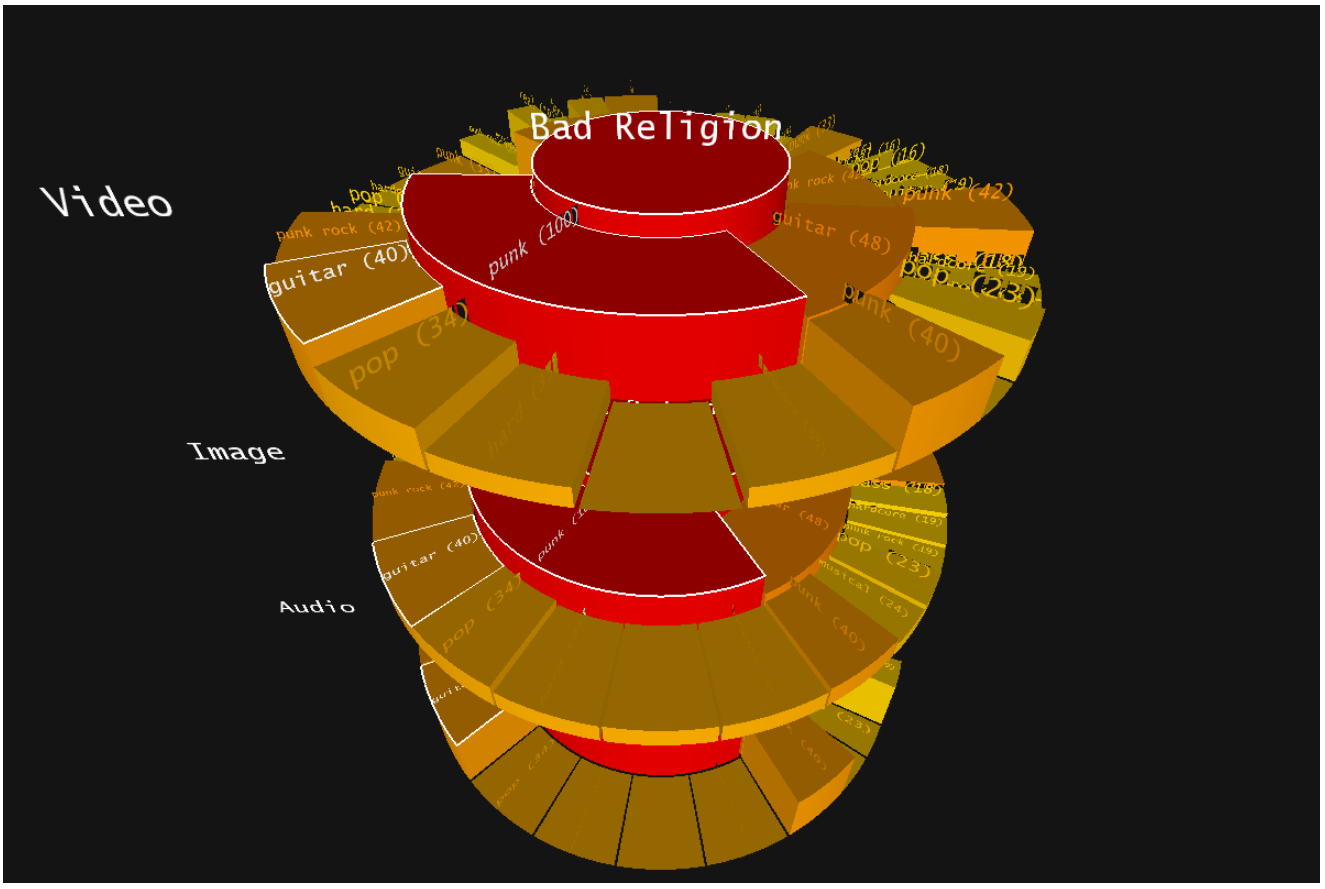


Figure 1.2: User interface of the *Co-Occurrence Browser*.

AGMIS

The *Automatically Generated Music Information System* (AGMIS) is the most important contribution among the applications developed in the context of this PhD thesis and will be elaborated in detail in Chapter 6. It offers implementations of the methods introduced in Subsection 1.3.1 and will be used to demonstrate their performance on a large set of artists. In addition to its Web page retrieval

and analysis functionalities and its information extraction and storage capabilities, AGMIS provides a Web-based user interface to conveniently access the extracted artist-related information. A sample screenshot depicting the search results for the band *Hammerfall* is shown in Figure 1.3. As for the relation between AGMIS and the other applications mentioned so far, AGMIS builds upon some Web retrieval functionalities provided by CoMIRVA, especially those for co-occurrence analysis and band member detection. A special version of COB is also included in AGMIS in order to provide a means of browsing the Web pages from which the artist-related information has been extracted.

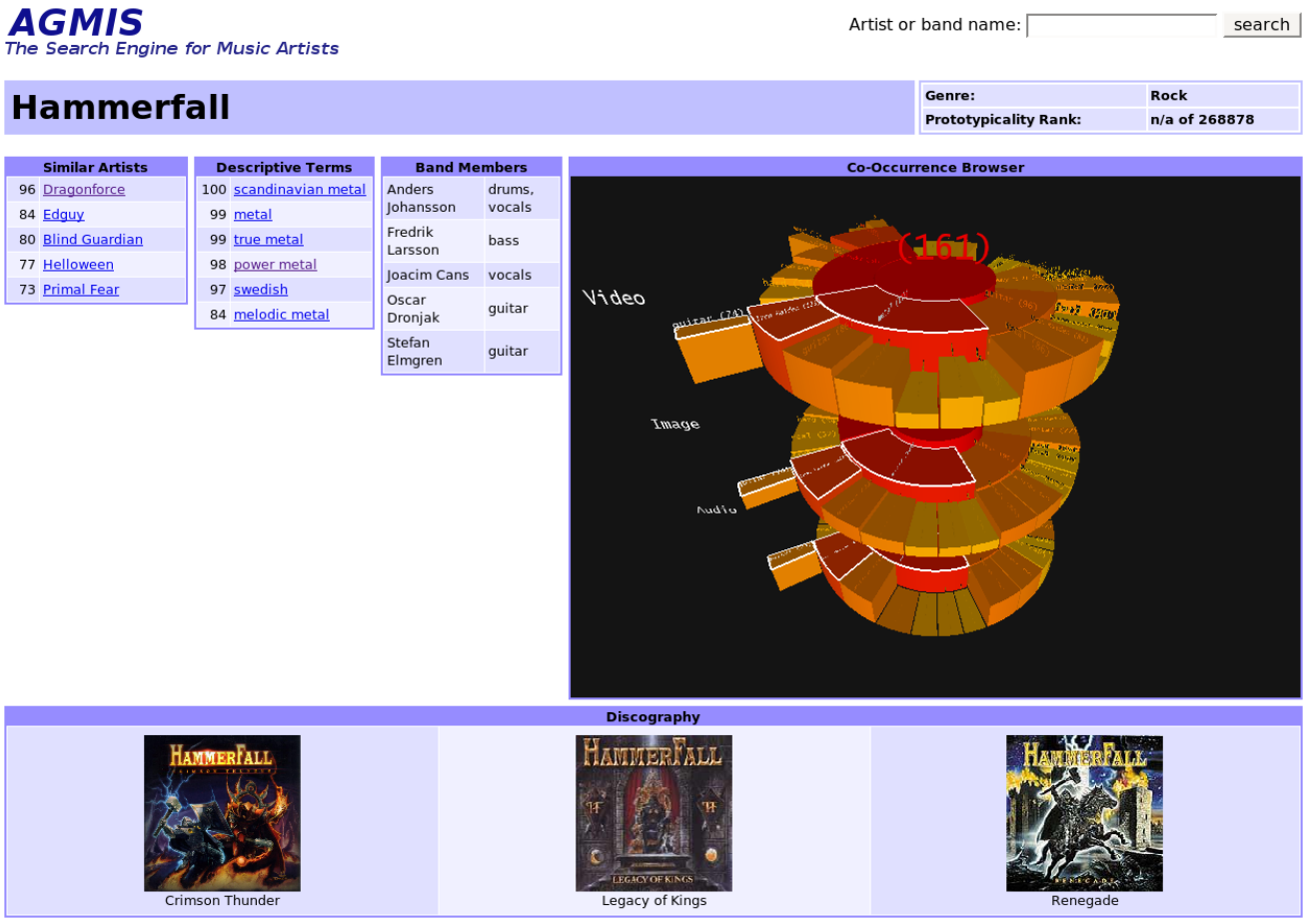


Figure 1.3: Web-based user interface provided by *AGMIS*.

The following two applications were co-developed (nepTune) or extended (Traveller's Sound Player) by the author. They are presented here to demonstrate the incorporation of artist-related information, which are automatically mined from the WWW, into MIR applications.

nepTune

The *nepTune* application provides a multimodal user interface to music repositories. By creating a virtual three-dimensional landscape based on audio features extracted from the user's music collection, nepTune offers a fun and intuitive means of accessing music – fun, because user access is performed in a game-like fashion; intuitive, because similar sounding pieces of music are clustered together, and these clusters are visualized as hills or mountains (depending on the number of pieces contained in the cluster). To get an impression of the user interface, the reader is invited to take a look at Figure 1.4. Besides the possibility of exploring the landscape via artist and track names, nepTune also offers a mode in which musically relevant terms are automatically extracted from artist-related Web pages and displayed on the landscape. In another mode, nepTune harvests images from artist-related Web pages and places them on the terrain. The main contribution to nepTune has been made by *Peter Knees*. However, the author of this PhD thesis also contributed considerably to the implementation, especially by adapting and integrating feature extraction and data projection algorithms from CoMIRVA and by implementing Web image retrieval functionality as well as some other minor components. For more details on the nepTune application, the reader is invited to consider [Knees et al., 2006b] or [Knees et al., 2007b].

Figure 1.4: *nepTune* user interface in the mode where artist and track names are displayed.

Traveller's Sound Player

The *Traveller's Sound Player*, developed by *Tim Pohle*, provides a user interface for mobile devices that organizes in a circular playlist the pieces of music in a collection. The playlist generation is performed in a way such that consecutive tracks show a high similarity according to some similarity measure which is applied to audio features extracted from the pieces of music. Generating the playlist is accomplished by applying a Traveling Salesman algorithm to the audio similarity matrix. The created playlist is then made accessible via a wheel that serves as a track selector. More details on the original version of the player, which has been described so far, can be found in [Pohle et al., 2005] and [Pohle et al., 2007a]. However, a shortcoming of this original version is that it does not offer the user any information on where on the wheel to find which kind of music, i.e., no metadata is displayed along the wheel. As a consequence, the user has to randomly select different angular positions with the wheel to find out about the music style of different regions of the playlist. To alleviate this problem, the author of this PhD thesis proposed in [Schedl et al., 2006c] a method to enhance the user interface by visualizing the distributions of certain categories of artist-specific metadata along the wheel. Such metadata, like genre or style information, are automatically mined from the Web by retrieving information on page counts and performing some simple statistical calculations. Regardless of the simplicity of the approach, it works surprisingly well. A screenshot of the extended application is given in Figure 1.5, where the genre distribution for *a capella* music is illustrated.

Figure 1.5: *Traveller's Sound Player* extended with the visualization of genre distributions.

1.4 Organization

The remaining chapters of this PhD thesis are organized as follows. In Chapter 2, an overview about the different types and categories of information that is suited to describe music artists is given and the

state-of-the-art approaches to obtaining each of the mentioned information categories are presented.
Chapters 3 and 4 then introduce the techniques elaborated for this thesis and compare them to existing methods. In detail, Chapter 3 illustrates the methods related to Web mining and information retrieval. It covers the retrieval and indexing of Web pages, as well as various possibilities of their analysis for the purpose of information extraction. Chapter 4 elaborates on the visualization of the two categories of artist-related information for which visualization techniques have been developed in the context of this thesis, namely artists that are prototypical for a certain subset of an artist collection (e.g., for a certain music genre) and the hierarchical data structure that is created by recursively performed co-occurrence calculation of terms on artist-related Web pages. The former information category is visualized in a graph-based fashion; for the latter one, a newly developed three-dimensional visualization approach is presented.
Chapter 5 is dedicated to the various experiments and evaluations that have been carried out to assess the quality of the developed methods. Wherever possible, quantitatively measurable evaluation methods were applied. However, to investigate the descriptiveness of terms (for artists or their music) and the usefulness of visualization approaches, qualitative, subjective assessments were made.
Chapter 6 describes the creation and implementation details of the core application of this thesis, the *Automatically Generated Music Information System* (AGMIS). This application serves to demonstrate the techniques elaborated in the context of this thesis by providing a Web-based user interface to access the artist-related information gained with these techniques.
In Chapter 7, the principal findings of this PhD thesis are summarized and conclusions are drawn. Finally, Chapter 8 points out possible research directions for future work.

Chapter 2

Categories of Artist-Related Information

In this chapter, the basic categories of artist-related data are presented. The principal aims hereby are to give an overview of the different concepts behind those categories, to elaborate on information that can be derived from raw Web data, to discuss related work, and to present the state-of-the-art in the corresponding research areas. Focusing on Web retrieval techniques will ensure staying within the scope of this thesis. Since a considerable number of related approaches have been developed or refined by the author of this PhD thesis or his colleagues, this chapter also serves as a technical introduction to Chapter 3, where more details on the developed or refined information retrieval approaches that are tailored to the problem of artist-related information gathering from Web pages will be given. Trying to broadly classify music artist-related information, the author proposes the three categories *relations between artists*, *descriptive artist properties*, and *additional information*, which are detailed in the following.

2.1 Relations between Artists

The first category of artist-related information can be described as a weighted, relational connection between artists ($a_1 \overset{w}{\leftrightarrow} a_2$). In the following, this relation will usually be a *function*. Hence, more formally, for two arbitrary artists a_1 and a_2 from an artist set A, there always exist an unambiguous value $w \in \Re$, such that $f(a_1, a_2) = w$. This function thus specifies the relationship between two artists according to some attribute.

2.1.1 Artist Similarity

In the case where this attribute relates to or describes the similarity between artists, $f(a_1, a_2)$ represents a *similarity function* (aka *similarity measure*). Such similarity measures play a vital role in many application scenarios of MIR, e.g., in music recommendation and playlist generation applications.

The concept of music similarity in MIR has traditionally been defined on the track level using a similarity

function which is calculated on some kind of audio signal-based features. Considering the great importance and widely usage of this signal-based, track level similarity, the following lines will be devoted to this subject.

There exists a vast amount of literature on the associated topics of audio feature extraction and similarity measurement between pieces of music. In general, audio-based (high-level) approaches can be broadly categorized into methods to capture *rhythmic* aspects of the analyzed music, e.g., [Pampalk et al., 2002a], [Dixon et al., 2003], [Gouyon et al., 2004], [Dixon et al., 2004], and methods that gather spectral properties in order to describe *timbre*, e.g., [Foote, 1997], [Logan, 2000], [Logan and Salomon, 2001], [Aucouturier and Pachet, 2004], [Aucouturier et al., 2005], [Mandel and Ellis, 2005]. The latter commonly extract *Mel Frequency Cepstral Coefficients* (MFCCs), which coarsely describe the spectral shape of an audio signal. The recent trend in track level similarity measurement seems to be combining multiple features, either as a fusion of purely audio-based features, e.g., [Allamanche et al., 2003], [Pampalk et al., 2004], [Pampalk, 2006], [Pachet and Roy, 2007], [Pohle and Schnitzer, 2007], or as an enrichment of audio-based features with Web-based features, cf. [Knees et al., 2006a], or with high-level musical descriptors assigned via collaborative tagging, cf. [Aucouturier et al., 2007]. Although it has been shown that methods for audio-based feature extraction and similarity measurement perform well for various MIR tasks, they suffer from certain limitations. The most apparent is that such approaches evidently depend on the audio signal in the form of a digital music file. Moreover, they have been suffering from high computational costs for certain algorithms, like such for calculating MFCCs or *Gaussian Mixture Models* (GMMs). With more efficient implementations and higher computational power of current hardware, however, this problem has been alleviated. Another problem, revealed in [Aucouturier and Pachet, 2004], is the likely existence of a glass ceiling for the performance of purely audio-based similarity measures.

Alternatively, similarity information can be derived from various kinds of metadata, in particular, from community metadata or cultural features as defined in Chapter 1. One of the first approaches in this direction can be found in [Pachet et al., 2001], where radio station playlists and compilation CD databases (using *CDDB*[1]) are exploited to extract co-occurrences between tracks and between artists. From these co-occurrences, similarity measures on the track level and on the artist level are derived and evaluated on rather small sets of artists and tracks (below 100 items each) using similarity judgments by music experts from *Sony Music* as ground truth. The main finding is that artists or tracks that appear consecutively in radio station playlists or on CD samplers indeed show a high similarity.

[1]CDDB is a Web-based album identification service that returns, for a given unique disc identifier, metadata like artist and album name, tracklist, or release year. This service is offered in a commercial version operated by *Gracenote* [gra, 2008] as well as in an open source implementation named *freeDB* [fre, 2008].

Another source for cultural features are the zillions of available Web pages. Since the World Wide Web reflects the opinions of a large number of different people, interest groups, and companies, approaches to derive artist similarity from Web data incorporate a kind of "collective knowledge" and thus provide an important indication for the perception of music. Moreover, such approaches can be used independently of any audio signal. In the following, methods to derive artist similarity from Web data are presented in chronological order of their publication.

In [Cohen and Fan, 2000] Cohen and Fan apply collaborative filtering techniques on lists extracted from Web pages which are found by "exploiting commercial Web-search engines". They query *Altavista* [alt, 2008] and *Northern Light*[2] [nor, 2008] to obtain Web pages related to music artists. The results are then used for artist recommendation. As ground truth for evaluating their approach, Cohen and Fan exploited server logs of downloads from an internal digital music repository made available within the Intranet of *AT&T*. They analyzed the network traffic for three months, yielding a total of 5,095 artist-related downloads. Unfortunately, no information on the number of artists used for evaluation is given in the paper.

In [Whitman and Lawrence, 2002] Whitman and Lawrence extract different term sets (unigrams, bigrams, noun phrases, artist names, and adjectives) from artist-related Web pages. Based on term occurrences, individual term profiles are created for each artist. The overlap between the term profiles of two artists is then used as an estimate for their similarity. For evaluation, the authors compared these similarities to two other sources of artist similarity information, which served as ground truth. First, information on similar artists were extracted from AMG. The authors note, however, that the expert-based similarity judgments as provided by AMG tend to be strongly influenced by a subjective bias of the respective music editors. Moreover, different editors may use different criteria for choosing similar artists. As second ground truth, user collections from *OpenNap*, a music sharing service, were gathered and analyzed. This is motivated by the assumption that artists are similar if they frequently co-occur in user collections. The test collection used for evaluation comprised about 400 artists. Remarkable differences between the individual term sets could be made out. The unigram, bigram, and noun phrase sets performed considerably better than the other two sets, regardless of the utilized ground truth definition.

In [Ellis et al., 2002] Ellis et al. use the same artist set as in [Whitman and Lawrence, 2002] to compare artist similarity judgments obtained via a Web-based survey with various other similarity measures. In addition to the measures derived from the different term sets in [Whitman and Lawrence, 2002], Ellis et al. propose a transitive similarity function on co-occurring similar artists from the AMG data, which they call "Erdös similarity". More precisely, the similarity between two artists a_1 and a_2 is measured

[2] Northern Light, formerly providing a meta search engine, in the meantime has specialized on search solutions tailored to enterprises.

as the minimum number of intermediate artists needed to form a path from a_1 to a_2 using the similar artist relationships by AMG. As this procedure also allows to derive information on dissimilar artists (those with a high minimum path length), it can be employed to obtain a complete similarity matrix. Furthermore, an adapted version of the described measure is proposed. The so-called "Resistive Erdös measure" takes into account that there may exist more than one shortest path of length l between a_1 and a_2. Assuming that two artists are more similar if they are connected via many different paths of length l, the Erdös similarity is adapted accordingly. However, this adjustment did not improve the agreement of the similarity measure with the ground truth data from the Web-based survey.

Similar artist data from AMG is also used by Cano and Koppenberger in [Cano and Koppenberger, 2004] to create a similarity network with 400 nodes (artists). Furthermore, playlist co-occurrences of more than 48,000 artists extracted from *Art of the Mix* [art, 2008] are visualized by a second network. Some interesting properties of these artist similarity networks are revealed. First, each artist is only connected with a small number of other artists. In spite of this sparsity, both networks showed one large cluster of nodes connecting more than 99% of the artists in the case of Art of the Mix and about 96% of the AMG artists. Furthermore, the average shortest path between two artists is remarkably small (4.7 for AMG and 3.8 for Art of the Mix). So is the clustering coefficient that estimates the probability that two neighboring artists of a given one are connected themselves, i.e., given that artist a_1 is similar to a_2 and to a_3, the probability for a_2 and a_3 being similar is quite small (0.3 for AMG and 0.1 for Art of the Mix).

In [Zadel and Fujinaga, 2004] Zadel and Fujinaga investigate the usability of two Web services to derive information on artist similarity. More precisely, they propose an approach that, given a seed artist, retrieves a list of potentially related artists from the *Amazon* [ama, 2008b] Web service *Listmania!*. Based on this list, artist co-occurrences are derived by querying the *Google Web API*[3]. Thereafter, the so-called "relatedness" of each Listmania! artist to the seed artist is calculated as the ratio between the combined page count, i.e., the number of Web pages on which both artists co-occur, and the minimum of the single page counts of both artists.

In [Schedl et al., 2005a] the author of this PhD thesis presents a similar approach, independent of Zadel and Fujinaga. However, unlike the method presented in [Zadel and Fujinaga, 2004], Schedl et al. derive complete distance matrices from artist co-occurrences retrieved from Google. This offers additional information since it can also be predicted which artists are **not** similar. Such information is necessary, for example, when it comes to creating playlists that incorporate a broad variety of different music styles. Moreover, in [Zadel and Fujinaga, 2004] artists are extracted from *Listmania!*, which uses the database of the Web shop Amazon. The number of artists in this database is obviously smaller than

[3]Google no longer offers this Web API. It has been replaced by several other APIs, mostly devoted to Web 2.0 development.

the number of artist-related Web pages indexed by Google. For example, most local artists or artists without a record deal are not contained, which makes the approach of [Zadel and Fujinaga, 2004] unsuitable for such artists. The approach of Schedl et al. also differs in the applied normalization method (minimum of the single page counts for both artists in [Zadel and Fujinaga, 2004] vs. page count of the first artist in [Schedl et al., 2005a]).

An approach that uses similar Web mining techniques as [Whitman and Lawrence, 2002] on artist information retrieved from Web pages is presented in [Knees et al., 2004]. Knees et al. however do not use specific term sets, but create a term list directly from the retrieved Web pages and use the χ^2 *test* for term selection, i.e., to filter out those terms from the list that are less important to describe the genre under consideration. After this term selection the common term weighting technique *term frequency · inverse document frequency* (TF·IDF) is employed to weight the remaining words and subsequently create a weighted term profile for each artist. This approach is evaluated in a genre classification setting, cf. Subsection 2.2.2, using *Support Vector Machines* (SVMs) and *k-Nearest Neighbor* (k-NN) classification on a test collection of 224 artists from 14 genres. To perform k-NN classification, the Euclidean distances between the term profile weights of each pair of artists are calculated, which in turn yields an artist similarity measure.

2.1.2 Artist Prototypicality

Finding artists that define a music genre or style, or at least are very typical for it, is a challenging and interesting task. Information on such prototypical artists can be exploited to support users in finding music more efficiently, especially in unknown collections like, for example, those used in visual artist recommender systems, e.g., *musicplasma* [mus, 2008]. As prototypical artists are well known, also unexperienced music listeners are able to assign them to a particular genre or style and can use them as reference points to discover similar but less known artists.

First attempts to measure artist prototypicality have been made by the author of this thesis in [Schedl et al., 2005b]. In this paper, an approach to derive complete artist prototypicality rankings for a set of genres is proposed. Such rankings also allow for measuring the degree of artist membership in a particular genre, thus for defining to which extent an artist produces music of a certain style or genre. According to [Schedl et al., 2005b], the prototypicality of a music artist is strongly related to how often music-related Web pages refer to the artist under consideration. Building upon [Schedl et al., 2005a], where a similarity measure based on co-occurrences of artist names on Web pages is presented, the approach proposed in [Schedl et al., 2005b] benefits from the intrinsic asymmetry of this similarity measure. More precisely, a so-called "backlink/forward link ratio" is introduced

and used to calculate prototypicality rankings. The basic assumption underlying this approach is that it is more likely that Web pages about a less known artist mention a well known artist from the same genre that vice versa.[4] A study on a test collection of 224 artists from 14 genres (the same as used in [Knees et al., 2004]) showed that this backlink/forward link approach produces mostly reasonable rankings. However, artists that equal common speech words, like *Kiss*, *Bush*, or *Prince*, are usually overrated since such terms obviously occur very often on artists' Web pages without referring to the artist, but to denote the common speech word.

To alleviate this problem, in [Schedl et al., 2005c] a correction for artists with extraordinary popularity is introduced. To this end, the ranking function is extended with a penalty term. This term downranks artists that show extremely high prototypicality values over all genres. Since even very popular and important artists are unlikely to be prototypes for all genres, extending the original genre prototypicality estimation proposed in [Schedl et al., 2005b] with such an "inverse overall prototypicality" factor considerably improved the results on the 224-artist-collection.

A comprehensive evaluation of different strategies for artist prototypicality estimation can be found in [Schedl et al., 2006b]. In this paper, three approaches are confronted – the two presented in [Schedl et al., 2005b] and [Schedl et al., 2005c] and a third one which is solely based on page counts returned to Google requests of the form "*artist name*"+"*genre name*". For evaluation, a collection extracted from AMG, comprising 1,995 artist names from 9 genres, is used. The ground truth is defined by the so-called "tiers", i.e., a very coarse[5] ranking of artists performed by AMG's editors. This ranking, which is created for each genre, should reflect the importance, quality, and relevance of each artist for the corresponding genre – cf. [amg, 2008]. The performed assessments showed a slight but significant dominance of the approach that penalizes exorbitant popularity over the other two approaches. However, as the ground truth data only consisted of three distinct ranking classes, the subjective improvements via downranking artist names that equal common speech terms was only partly reflected by the evaluation results. Indeed, when looking at the 10 highest ranked artists for each genre, the qualitative differences between the penalizing approach and the other two seemed larger.

2.2 Descriptive Artist Properties

As spoken and written words serve as principal means of human communication, it is natural to describe music artists in this vein. In the following, two important text-based information categories that are frequently used in MIR research are discussed. First, the quite general group of *descriptive terms* is

[4] To give an example, it is more natural to say that the Finnish heavy metal band *Sentenced* sounds like the well known pioneers *Metallica* than vice versa. Hence, it is reasonable to assume that Web pages about *Sentenced* frequently contain phrases like "The band was influenced by Metallica".

[5] No more than three tiers are distinguished.

elaborated. The subsequent subsection addresses the concepts of musical *genre and style*, which are indispensable for various MIR tasks, but also frequently criticized.

2.2.1 Descriptive Terms

Descriptive terms in the context of MIR refer to denotations of specific properties of artists/bands or their music. Such properties may include moods, emotions, tempo, usual listening situations, instruments, geographic locations related to song or artist, creative periods of the artist, epoch in which the song was released, styles, and genres. The latter two have gained much importance, especially for the evaluation of MIR approaches, and will thus be discussed separately in the next subsection.

Since today's largest source of textual information is most probably the WWW, it is no surprise that many MIR approaches that aim at assigning descriptive properties to an artist exploit the Web. This is usually performed either by processing *static Web pages returned by a search engine* or by analyzing *user generated content* that is created by collaborative tagging.

Among the former category are some of the already presented approaches to similarity measurement, e.g., [Whitman and Lawrence, 2002], [Knees et al., 2004], which involve the creation of artist-related term profiles. Such profiles obviously contain descriptive terms. Hence, the corresponding approaches address the information category covered in this subsection. However, as those approaches do extract (nearly) all terms from a given set of Web pages, even after term weighting and selection, the resulting term set overall shows a quite unspecific nature in regard to the terms' descriptiveness for the artist under consideration. After all, it must be stated that the mentioned approaches do not claim to automatically annotate artists with descriptive terms, instead their main intention is to derive artist similarities.

Addressing the large amount of terms not directly related to the artist or his/her music, as mentioned above, in [Pampalk et al., 2005] Pampalk et al. use a dictionary of about 1,400 terms taken from various sources and chosen according to the authors' judgment of their usefulness to describe music. This paper focuses on the automatic *hierarchical clustering* and *description* of artists. The former is achieved by training a hierarchical version of the *Self Organizing Map* (SOM), cf. [Kohonen, 2001], [Dittenbach et al., 2002], the latter by applying different term weighting techniques. The proposed approach again relies on Web pages retrieved via artist-related requests to Google. One of the main findings of a comparative study conducted in [Pampalk et al., 2005] is that considering only the terms in the dictionary for term weighting and subsequent clustering outperforms using the unpruned version of the complete set of extracted terms, with respect to the suitability of the terms to describe artists or clusters of artists.

Knees et al. in [Knees et al., 2006b] use a similar dictionary of musically relevant terms to label a

three-dimensional representation of a SOM that is trained on signal-based features extracted from audio files. In this case, only the result page returned by Google for queries for artist names is indexed using the dictionary, whereas most other approaches retrieve and analyze the complete Web pages. This considerably reduces network traffic and runtime complexity while still yielding adequate results.
In [Schedl et al., 2007b] Schedl et al. likewise use a dictionary to hierarchically organize a set of artist-related Web pages. Besides the different focuses and application areas of [Pampalk et al., 2005] and [Schedl et al., 2007b], the two approaches also differ in that Schedl et al. do not use a SOM for clustering, instead they hierarchically organize a set of Web pages by alternately performing term weighting and creating subsets of the Web pages according to co-occurring terms.
Another, very straightforward, approach to assign textual properties to a given music artist is followed in [Schedl et al., 2006c]. This approach is similar to the one presented in [Schedl et al., 2005a] in that it also makes use of co-occurrences derived from artist-related Web pages. The core idea of the approach is to estimate the conditional probability for the artist name under consideration to be found on a Web page containing a specific descriptive term and the probability for the descriptive term to occur on a Web page known to contain the artist name. To this end, a set of predefined genres and other attributes, like preferred tempo or mood of the artist's performance, is used. The aforementioned probabilities are then calculated, and the most probable value of the attribute under consideration is assigned to the artist.
In [Geleijnse and Korst, 2006] Geleijnse et al. propose an approach for artist categorization that is very similar to the one presented in [Schedl et al., 2006c]. Indeed, the two approaches only differ in the employed normalization method.[6] In addition, [Geleijnse and Korst, 2006] proposes two other approaches, both also making use of term co-occurrences. The second approach retrieves the Google page counts returned for exact phrase searches of the form "*g* artists such as *a*", where *g* represents the genre and *a* the artist. These page counts are thereafter used to predict the genre of an artist. The third approach does not rely on page counts returned by the search engine, but instead fetches the top-ranked Web pages returned for queries for the artist name or attribute value. From the set of Web pages returned for artist queries the term frequencies of the attribute values are extracted; on the search results for the attribute values the term frequencies of the artist names are calculated. Both term frequencies are combined for each combination of *a* and *g*, and again used to predict the requested artist property. The three approaches are also complemented with artist similarity information derived by a technique very

[6] Normalization is necessary to account for the heavily varying total number of Web pages containing the attribute value or artist name under consideration. Schedl et al. divide the combined co-occurrence count (aka page count), i.e., the number of Web pages containing both artist name and attribute value, by the total number of Web pages containing the artist name or the attribute value (formally, $\frac{pc_{a,g}}{pc_g}$ or $\frac{pc_{a,g}}{pc_a}$, where a is the artist name and g is the attribute value), whereas Geleijnse et al. divide the combined co-occurrence count by the sum over all artists' combined co-occurrence counts (formally, $\frac{pc_{a,g}}{\sum_{b \in A} pc_{b,g}}$, where A is the set of all artists under consideration, a and g are defined as above).

similar to [Schedl et al., 2005a]. Evaluation is performed on the tasks of categorizing artists according to genre and according to mood using the 224-artist-collection from [Knees et al., 2004]. Good results are achieved for the genre categorization task, whereas the results for categorizing artists into moods are not satisfying. This seems in accordance with the findings of [Schedl et al., 2006c].

Another MIR-related application area that builds upon methods for term extraction from Web pages, term weighting, audio feature extraction, and similarity measurement are music search engines of the kind of [Knees et al., 2007a]. In this paper, Knees et al. relate audio features calculated on a given music collection with terms extracted from Web pages that contain some or all of the metadata present in the music collection (specifically, artist, album, and song names). These terms are then weighted using an adapted version of the TF·IDF measure and joined with the audio features to build a feature vector for each track, which serves as a track descriptor. More precisely, the term weights of each track descriptor are modified by incorporating the term weights of acoustically similar pieces of music using a Gaussian weighting. In cases where no usable Web information on the track level can be made out, i.e., there exists no Web page containing the exact title of the track under consideration, this feature combination nevertheless ensures a valid track descriptor. The described approach allows for searching music collections via descriptive natural language terms, e.g., by issuing queries like "guitar riff" or "metal band with front woman". After the user has entered such a query, a similarity score between the query and the individual track descriptors is calculated, and a sorted list of matching tracks is output.

A somewhat similar, but simpler, system is presented in [Celma et al., 2006], where Celma et al. propose a music search engine that crawls audio blogs via *RSS feeds*. These feeds contain links to music files as well as short textual descriptors of the pieces. The textual descriptors are combined with metadata extracted from the audio files (in particular, from the *ID3 tags*), which enables to match textual user queries with music files. Moreover, for each suggested song, a list of similar songs according to signal-based similarity calculations can be displayed. This music search engine is accessible via [sea, 2008].

In the era of the "Web 2.0", static Web pages are more and more frequently replaced by dynamic, user generated content. In particular, a process known as *collaborative tagging* is commonly applied to generate metadata in the form of *tags*. To this end, users collaboratively annotate a specific item or entity by descriptions, usually chosen by themselves. This collaborative tagging can obviously be applied to describe music. The various tags assigned to artists, albums, and sometimes even tracks by the users of last.fm prove it. As such tags are a valuable source of information, some recent publications make use of them.

For example, in [Pohle et al., 2007b] Pohle et al. construct artist-related term sets from the Web and from last.fm tags for nearly 2,000 artists gathered from AMG. As for feature extraction, the artist infor-

mation gathered from Web pages is represented by TF·IDF vectors, whereas from last.fm the weighted user tags assigned to each artist are retrieved. Using these two artist representations, the authors subsequently apply *non-negative matrix factorization* (NMF), cf. [Lee and Seung, 1999], to cluster the artists similarly to [Xu et al., 2003b]. To this end, NMF decomposes the data into a fixed number of concepts[7], each of which is represented as a weighted combination of the feature terms. The most important terms for each concept are used to describe the corresponding cluster in a visualization. An interesting finding of [Pohle et al., 2007b] is that a large part of the most important terms of each concept is clearly related to a particular genre. After performing the NMF, each artist is represented by a 16-dimensional vector in the concept space. This representation is furthermore used for artist recommendation. To this end, similarities between user defined weights for each concept as given by sliders in a user interface and the 16-dimensional artist representations in the concept space are calculated, and the best matching artists are recommended to the user.

Geleijnse et al. propose a method to dynamically create ground truth datasets for artist classification tasks from last.fm tags in [Geleijnse et al., 2007]. To this end, artist level tags and tags describing the top-ranked tracks of the artist are considered. Since this tag data usually contains much noise, it is cleaned up by running through various filtering steps. Among other things, stemming is performed to reduce, for example, the terms "Hip-Hop", "hip hop", and "hiphop" to a canonical representation. Moreover, tags that equal the artist name are removed as well as obvious spelling mistakes. To this end, all tags that do occur very rarely in the complete tag set for the 1,995 artists analyzed in the paper (the same artist set as in [Schedl et al., 2006b] was used) are removed. An interesting finding is that although users can chose arbitrary terms as tags, the number of frequently used tags is relatively small. Indeed, the number of tags applied to at least 5% of the artists is in the order of hundreds, whereas the number of unique tags for the complete artist collection approaches 15,000. Another result of the analysis is that the most popular tags commonly represent descriptive artist properties related to genre, style, or epochs. However, also less discriminative terms like "favorites", "good", or "seen live" are often encountered. The authors also evaluated the consistency of the tags by calculating the number of overlapping tags of similar artists and of randomly chosen artists. Their conclusion is that the number of tags shared by similar artists is indeed much larger than the number of tags shared by randomly chosen artists.

In [Hu et al., 2007] Hu et al. address tags that describe moods. The authors extract tags from last.fm at the song level in order to create a ground truth set for automatic mood categorization tasks. The resulting tag list is then processed by a part-of-speech (POS) tagger to retain only adjectives. From the most frequently applied tags that remained, 19 tags describing moods were manually selected.

[7] For the data set used in [Pohle et al., 2007b], 16 clusters/concepts seemed reasonable.

Furthermore, Hu et al. cluster a collection of 2,554 tracks according to these 19 tags.
An approach to automatically assign tags to a given song is proposed in [Eck et al., 2007]. This is achieved by using a machine learning algorithm which is trained on a set of signal-based features and last.fm tags extracted at the artist level. Although the artist set used in the experiments comprised 50,000 items, the tag set used for training the classifier was restricted to 13 popular genre names.

2.2.2 Genre and Style

Music genre and style are very widely used concepts to describe music, commonly on the artist or album level. From a general and abstract point of view, the concept of genre is used to subsume a set of artists that share some common properties[8] and to differentiate those artists which do not show these properties. Thus, the concept of genre has a strong categorizing function. It is used by the music industry to broadly categorize their artists under contract as well as by the billions of listeners to describe the kind of music an artist performs. Furthermore, the concept of genre obviously plays a vital role in how users organize their music collections, since the most commonly applied structurisation scheme is still that of `genre-artist-album-track`.
Traditionally, the different music genres were shaped by music retailers in order to offer their customers an efficient way to quickly find their desired music in record shops. However, with the advent of digital music distribution via the Web and the resulting boost in the amount of music offered at one – now virtual – place, more complex genre taxonomies emerged. Such taxonomies, like those used by *Amazon* [ama, 2008b] or *mp3.com* [mp3, 2008], typically comprise hundreds of genres and subgenres. Despite this prevalent usage of genre information to categorize music, or actually because of it, even commonly used genres are often interpreted in different ways. As a result, most genre taxonomies are inconsistent. Aucouturier and Cazaly performed interesting studies on this topic and presented them in [Pachet and Cazaly, 2000]; a summary can be found in [Aucouturier and Pachet, 2003]. More precisely, the authors analyzed and compared existing genre taxonomies, some used by the music industry, others extracted from music-related Web sites. Their main findings were the following:

- There is no consensus in the genre names, i.e., the used genre labels barely overlap between the analyzed taxonomies.
- There are large differences in the hierarchical structure of the various taxonomies, even though the total number of genres is in the same order of magnitude.
- Not even general and widely used genre names, like "Rock" or "Pop", have consistent definitions,

[8] Such properties are usually related to their music, e.g., instrumentation or playing techniques, but may also describe their geographical origin.

i.e., the corresponding sets of artists overlap only slightly between the different taxonomies.

- The meaning of the genre labels is often unclear. For example, does a genre "World Italien" contain all artists from Italy regardless of their music style, or all artists that sing in Italian regardless of their nationality? What about the difference between genres labeled "Pop Metal" and "Metal Pop"?

- The semantics of the hierarchical connections (genre–subgenre) is neither always clear nor consistent. This relation may have a geographical, temporal, or specifying/detailing meaning, or sometimes just serve as an aggregator (e.g., RnB/Soul–RnB and RnB/Soul–Soul).

In [Aucouturier and Pachet, 2003] Aucouturier and Pachet discuss different genre categorization strategies and describe approaches to infer genre information based on manual annotation, signal-based classification, and clustering according to similarities derived from various metadata.
The author of this PhD thesis presents in [Schedl et al., 2006c] an approach to derive genre information from page counts provided by Google. These page counts are used to estimate the relatedness of an arbitrary artist to each of a set of genres. Since the presented approach predicts the genre of a given artist in a probabilistic fashion, it is possible to overcome the problem of artists that cannot be assigned a unique genre, but instead combine elements from various genres or change their style remarkably over time. Independent of Schedl et al., Geleijnse and Korst present basically the same approach in [Geleijnse and Korst, 2006].
In spite of the inconsistent definitions of music genres, classification of artists or songs into genres is nevertheless one of the most frequently employed evaluation methods for newly proposed approaches to feature extraction and similarity measurement. Examples can be found in [Tzanetakis and Cook, 2002], [Xu et al., 2003a], [Burred and Lerch, 2003], [Gouyon et al., 2004], [McKay and Fujinaga, 2004], and [Knees et al., 2004]. This is understandable to a certain extent as alternative evaluation approaches, e.g., user studies, are much more laborious.
As it has been shown above, genre is an ill-defined concept. The same holds for style. In many scientific publications, the term "style" is used as a substitute for "genre" or "subgenre", in the latter case, to describe more precisely the kind of music an artist performs. On the other hand, some works, like [Moore, 2001], regard style as being much more influenced by aspects like subject matter and, thus, vote for a clear distinction between the two concepts. Without going too much into detail here, it can be stated nevertheless that there is no general agreement on what really makes a music genre or style. In the context of this thesis, the term "genre" will be used to describe a general category of music, whereas "subgenre" and "style" will be used interchangeably to denote a more specific distinction.

2.2.3 Band Members and Instrumentation

On the level of a music band, an important information is that of their members and the instrument(s) they play. Not only is such information interesting for deriving band and artist histories, but it may also serve as an additional dimension to enrich similarity measures or to build relationship networks. The former aspect is strongly related to automatic biography generation, e.g., [Alani et al., 2003], whereas the usage of membership information for similarity measurement is motivated by different music styles that are often caused by changes in the line-up of a band.

First steps towards automatically detecting music band members and instrumentation using Web content mining techniques were made in [Schedl et al., 2007c] by the author of this PhD thesis. The presented approach basically comprises the four steps *Web retrieval*, *named entity detection*, *rule-based linguistic analysis*, and *rule selection*. First, Web pages that potentially contain information about band members are sought using Google. Second, in the named entity detection step, N-grams are extracted from these Web pages, and some basic filtering of N-grams that are very unlikely to represent artist names is performed. This yields a set of potential band members. Third, these potential members and their surrounding text are investigated for matches with a set of rules formulated in natural language.[9] Finally, for each (member, instrument)-pair of the band under consideration, the rule with the highest number of matches is selected, and for each instrument, the respective member is predicted.

Although this approach yields remarkable results, especially when taking into account that not all correct band members occur in the set of Web pages retrieved for the band under consideration, predicting only one member per instrument certainly does not reflect the line-up of many bands. Hence, in [Schedl and Widmer, 2007] an adaptation of the approach as well as a more comprehensive evaluation are presented. More precisely, the number of matches is aggregated over all rules for each (member, instrument)-pair, and those pairs whose aggregated match count falls above a certain threshold are predicted. This refinement allows for an `m:n` assignment between band members and instruments. The predominance of this modified approach over the one proposed in [Schedl et al., 2007c] is also reflected by the experiments conducted in [Schedl and Widmer, 2007].

2.3 Additional Information

2.3.1 Song Lyrics

The lyrics of a song represent an important aspect of the semantics of music since they usually reveal information about the artist or the performer: e.g., cultural background (via different languages or use

[9] Typically used rules are, for example, "M plays the I" or "M is the R", where M is the member, I is the instrument, and R is the role M plays within the band.

of slang words), political orientation, or style of music (use of a specific vocabulary in certain music styles). In spite of this considerable amount of information encoded in a song's lyrics, not much scientific work has been devoted to analyzing lyrics. The few works that tackle the topic are summarized in the following.

A first approach to automatically retrieving song lyrics from the Web can be found in [Knees et al., 2005]. In this paper, requests to a search engine are used to obtain a set of potential song lyrics and an algorithm from the field of bioinformatics, more precisely a hierarchical version of *multiple sequence alignment*, cf. [Corpet, 1988], is employed to overcome the problem of eliminating erroneous lyrics and to construct the most probable version of a song's lyrics. In [Korst and Geleijnse, 2006] this approach has been speed up considerably while at the same time almost retaining the quality of the output. This is achieved by removing text fragments which are considered outliers at an early stage of the lyrics processing and by considering the structure (HTML tags) of the retrieved Web documents that potentially contain the song lyrics.

In [Logan et al., 2004] a set of 16,000 song lyrics was used to determine artist similarity. To analyze and compare the semantic content of the lyrics, probabilistic latent semantic indexing was applied.

In [Mahedero et al., 2005] various information retrieval techniques are applied to perform language recognition, structure extraction (categorizing the parts of a song's lyrics into the classes *intro*, *verse*, *chorus*, *bridge*, and *outro*), thematic categorization (classifying the lyrics into the 5 distinct classes *love*, *violent*, *protest (antiwar)*, *christian*, and *drugs*), and similarity measurement based on song lyrics. Although the reported results are still improvable, [Mahedero et al., 2005] illustrates some interesting aspects that are encoded in a song's lyrics.

Apart from scientific research on semantic lyrics analysis, supportive and convenient applications that assist users in the retrieval of lyrics are freely available on the WWW. The most notable among them is *EvilLyrics* [evi, 2007]. EvilLyrics cooperates with the most popular music players and searches for lyrics while a song is played. Given artist and track name, a set of known lyrics portals are searched via a common search engine, the resulting Web pages are parsed, and the user is finally presented with the bare lyrics and can choose between multiple versions from different Web pages. Although EvilLyrics is a handy and speedy tool, it suffers from two major drawbacks: First, the retrieval is limited to a set of known lyrics portals, making it infeasible to exploit pages focusing, for example, only on lyrics from one particular artist, like official artist pages or fan pages. Second, finding the correct version of a song's lyrics is left to the user.

To summarize, automatically retrieving, analyzing, and interpreting song lyrics are all challenging tasks on their own and still require considerable scientific effort to be solved. However, since lyrics are a source of interesting semantical metadata about a song or an artist, research into this directions is

worth pursuing.

2.3.2 Discography and Album Covers

The concept of the music album, albeit often said to be antiquated in the era of digital distribution of individual songs, nevertheless has not lost much of its popularity among music lovers and collectors. Nearly all music releases are still effected in this format, and for the majority of consumers it still seems more appealing to have the complete album than just a few selected tracks.

Discography information, i.e., information on all albums released by a specific artist or band, hence plays a vital role for identification. An important part of this discography information is the album cover artwork. It is essential to recognize an album and also serves as a stylistic device. Bearing this in mind, remarkably few scientific effort has been made to automatically gather album cover artwork or integrate it in MIR applications.

As for the former task of *automatically retrieving album covers*, two categories of approaches can be distinguished: the ones that make use of *album cover databases* and the ones that *crawl the Web* to find cover images. Access to cover image databases is provided, for example, by specialized Web sites like *CoverUniverse* [cov, 2007]. Also online stores that distribute digital music, like *Amazon* or *Wal-Mart*, usually maintain such databases. Although these Web sites frequently offer high quality scans of cover images, the number of available covers is usually rather small compared to the number accessible by crawling the Web. This drawback is somewhat alleviated by programs such as *Album Cover Art Downloader* [alb, 2008], which combine the search results from different album cover databases. However, the possibly large number of potential covers that comes along with this approach raises the problem of how to select the correct cover image. Therefore, such programs only perform semi-automatic retrieval, i.e., the user still has to select the most appropriate image from a set of candidate covers.

The second category of approaches, those that make use of Web mining techniques, to the best of the author's knowledge solely comprise the one proposed in [Schedl et al., 2006a]. In this paper, Schedl et al. automatically retrieve album cover artwork via querying different image search engines and applying various content-based filtering techniques on the returned sets of images in order to determine **one** image that is the most likely album cover.

As for the *usage of album cover artwork* in MIR applications, over the past few years, enriching digital music player software with images of album covers has been becoming more and more common. For example, the Linux music player *Amarok* [ama, 2008a] integrates a function to automatically download cover images from Amazon since 2004. *Apple* introduced in 2006 a technique called *CoverFlow* [cov, 2008] in its music player software *iTunes*, and later also in its mobile music players. CoverFlow provides a user interface to browse a music collection by rummaging in album covers. Other

companies and free software developers followed.

The few scientific work on the usage of album cover artwork in MIR research include [Brochu et al., 2003], where Brochu et al. build a multimodal mixture model that integrates features derived from color histogram representations of album cover images, from song lyrics, and from musical scores. Their approach allows for querying multimedia databases via images, text, and music as well as for clustering a set of songs based on the created mixture models. Another approach that combines music and images is presented in [Bainbridge et al., 2004], where Bainbridge et al. use album covers among other images as data source for applying collaging techniques. The authors present a user interface which aims at facilitating browsing digital music libraries by giving both aural and visual feedback.

Chapter 3

Techniques Related to Web Content Mining and Information Retrieval

This chapter elaborates on the techniques for extracting the pieces of information that represent the concepts introduced in the last chapter. The topics dealt with are the *retrieval of music-related Web pages*, the subsequent *indexing of the Web pages' content*, and finally, the *extraction of the desired information*.

3.1 Web Page Retrieval

The first task that has to be addressed when searching for information about music artists on the Web is the discovery of Web pages that contain relevant information in the zillions of pages available on the Web. To this end, two different strategies can be pursued – using a *focused crawler* or *querying a search engine with constrained queries*.

A focused crawler, as proposed in [Chakrabarti et al., 1999], takes as input a set of seed URLs which point to Web pages that are relevant to the topic(s) requested by the user. In an iterative training process involving user feedback, a probabilistic classifier then learns which pages are related to which topic according to a given taxonomy. Starting with the links present on the seed pages, the focused crawler subsequently follow the links that are judged highly relevant for the requested topic(s) by the classifier. Regarding the Web pages as nodes of a graph and the links between them as the graph's edges and starting at the nodes representing the seed pages, the focused crawling approach by Chakrabarti et al. employs a *best-first search* strategy, e.g., [Russell and Norvig, 2003a], using the relevance judgments by the classifier as heuristic to successively explore the space of Web pages in a topic-directed manner. Since its proposal in 1999, focused crawling has experienced considerable refinements. Some of these refinements are outlined in [Bergmark et al., 2002] and include, for example, adapting the heuristic used to chose a link to follow in order to account for the fact that sometimes relevant pages are "hidden" between a number of irrelevant ones and would therefore have no chance to be found by the locally constrained best-first search algorithm employed in [Chakrabarti et al., 1999]. Another interesting refinement can be found in [Xu and Zuo, 2007], where Xu and Zuo model properties of Web pages and relations between them in *first-order logic* and use a relational learning algorithm to

derive first-order rules, which guide the crawling process.

An alternative way to obtain Web pages about a certain topic is to rely on the results returned to queries to a Web search engine. A large number of authors pursue this approach. In the context of MIR, querying search engines to gather sets of music-related Web pages is performed, for example, in [Cohen and Fan, 2000], [Whitman and Lawrence, 2002], [Knees et al., 2004], [Knees et al., 2007a], and [Geleijnse and Korst, 2006]. More general approaches, which are not related to MIR, are presented, for example, in [Cimiano and Staab, 2004] and [Cimiano et al., 2004]. Approaches that retrieve information by querying a search engine, but take only the returned *page counts* into account can be found, for example, in [Schedl et al., 2005a], [Schedl et al., 2006c], [Cimiano and Staab, 2004], and [Cimiano et al., 2004]. The main difference of the search engine-based approaches to the strategy of focused crawling is that major search engines use Web crawlers that usually employ a *breadth-first search* algorithm, e.g., [Russell and Norvig, 2003b], to explore the Web. Since such crawlers aim at visiting as many Web pages as possible, the main challenge when using queries to a search engine for Web page selection is to restrict the search results to pages related to the desired topic. This is commonly addressed by enhancing the search query with additional keywords. For Web mining in MIR, [Whitman and Lawrence, 2002] proposes confining the search by the keywords "music" and "review", in addition to the artist name under consideration. The aim is to direct the search towards album reviews. This query scheme was shown to be successful for genre classification tasks.

The techniques elaborated in the context of this PhD thesis also benefit from the large amount of Web pages crawled by major search engines as the author follows the respective approach to Web page retrieval. As for the query refinement in order to direct the search to Web pages related to music, different query schemes tailored to the information category under consideration are used. The keyword "music" is added to the search query by default, to avoid problems with artist names that equal common speech words, like *Bush*, *Prince*, or *Kiss*. To give an example, the complete query scheme for a search aimed at finding the members of a particular band could look like `"artist name"+music+members`. More information on the used query schemes will be given in Subsection 5.2. Regardless of the employed strategy (focused crawling or search engine), the determined Web pages are subsequently fetched, e.g., using tools such as *wget* [wgt, 2007].

3.2 Indexing

After having downloaded the content of the Web pages, the next step is to build an *index* of the collection of Web pages. A comprehensive elaboration on various aspects of creating, maintaining, storing, and querying/searching in file indexes is given by Zobel and Moffat in [Zobel and Moffat, 2006].

Generally, an index is a data structure that maps terms occurring in a collection of documents, e.g., Web pages, to a list of document identifiers.[1] There exists, in principal, two different variants of indexes with respect to the mapping they describe. The mapping may either give, for each term t, a list of documents in which t occur or a list of documents and the precise positions of t within each document. In a regular expression notation, the two variants of the mapping can be written as follows.

DLI: `term` $\mapsto$ `document*`

WLI: `term` $\mapsto$ `(document,position+)*`

Unfortunately, there is no agreement on the denomination used for each of the two variants. The former is often referred to as *document-level inverted index*, *record-level inverted index*, *inverted file index*, or just *inverted file*, while the latter may be named *full inverted document index*, *word-level inverted index*, *full inverted index*, or *inverted list*. The major advantage of a full inverted document index is that it allows for *phrase search*, i.e., finding an exact phrase within a document.

As indexes tend to occupy substantial amounts of disk space[2], compression is vital. An easy, albeit efficient, means is to encode the document identifiers given as numbers using *d-gaps*. By this means, given the sorted sequence of identifiers, only the first identifier is stored as absolute number, whereas all following ones are stored as offsets relative to the preceding one. For example, instead of storing the sequence <5, 8, 9, 12, 33>, the sequence <5, 3, 1, 3, 11> is recorded. This transformation, combined with *variable-length encoding* of integers, e.g., [Golomb, 1966] or [Elias, 1975], can significantly reduce the total index size.

As for the questions of how a document is parsed and which of its terms are actually indexed, there are basically three frequently performed preprocessing steps to the indexing that address these issues. The first one is *casefolding*, i.e., the conversion of all characters into lower case. On the one hand, casefolding saves space as words that only differ in the use of upper and lower case are not stored twice. However, case sensitive searches are obviously infeasible when using an index that was built on casefold text. Moreover, techniques like *named entity detection*, cf. Subsection 3.3.4, rely on case information. The second preprocessing technique is *stopping* or *stop word removal*. Stopping removes those terms from the input documents that do so frequently occur in natural language that they barely bear any relevant information, e.g., “a”, “the”, “of”, “in”, “to”, or “who”. Including such terms in the index significantly slows down the indexing process. On the other hand, for some tasks, like the extraction of *Hearst patterns*, cf. Subsection 3.3.4, indexing of all terms is required. Lists of common stop words for

[1] In the following, the terms “document” and “Web page” will be used interchangeably as in the methods developed, documents are always Web pages.

[2] According to [Zobel and Moffat, 2006], typical indexes require between 20% and 60% of the size of the original data.

different languages can be found, e.g., in [sto, 2008]. The third parsing approach, known as *stemming*, maps inflected words to a canonical representation – their stem. This representation does not require to correspond with the morphological root of the word. However, the stemming algorithm must ensure that related words are mapped to the same stem. To give an example, the terms "replacement", "replace", "replaced", and "replacing" may be mapped to the common stem "replac". A widely used stemming algorithm for the English language is *Porter stemming* [Porter, 1980], [Porter, 1997]. It has been slightly adapted and improved in the context of the *Snowball* project [sno, 2008], which provides a framework for writing stemming algorithms as well as stemmers for several languages. In the context of this PhD thesis, different combinations of parsing strategies were pursued, depending on the task addressed.

If the documents to be indexed are Web pages, another issue is to decide on the indexing of terms within HTML tags and of the tags themselves. In the field of Web content mining, neither is usually performed. Except for the search for multimedia content and for album cover artwork, which will be addressed later, HTML tags are ignored by the indexing performed in the context of this thesis.

For the conducted experiments that will be described in Chapter 5, simple indexers implemented in Java by the author were used. However, for the large amount of Web pages gathered to build AGMIS, cf. Chapter 6, a more sophisticated indexer was required, especially in regard to data storage capabilities and flexibility. Therefore, the author opted for the open source indexer *Lucene Java* [luc, 2008], which was adapted for this purpose.

Music Dictionary

The usual motivation for building an index is to provide a means of efficient search for a variety of queries, as offered, for example, by common Web search engines. In this case, it is essential to allow for specifying the query using arbitrary search terms. In contrast, for the specific task of determining descriptive terms for a music artist, cf. Subsection 3.3.3, using a specialized term list focusing on words that are somehow related to the music or the artist under consideration seems to be better suited than building a large index comprising (nearly) all words[3] that appear in any of the input documents. To this end, a music dictionary, similar to the one used in [Pampalk et al., 2005], has been assembled by the author. It is based on various sources such as *Wikipedia* [wik, 2007b], *AMG* [amg, 2007a], and *Yahoo! Directory* [yah, 2007a] and contains terms that denote music genres and styles, epochs, instruments, moods, geographic locations, and other terms that are somehow related to music.

[3]The unique words eventually included in the index depend, of course, on the use of casefolding, stopping, and stemming.

Multimedia Content

As for indexing the multimedia content of Web pages, which is required for the user interface presented in Subsection 4.2.6, lists of common file extensions for music, image, and video files are gathered from [wik, 2007a]. Subsequently, the HTML code of each Web page is searched for links to files whose file extension occur in one of the lists. More precisely, the HTML attributes that typically contain such links are analyzed, and the respective file names are extracted.[4] Finally, the URLs of the detected multimedia files are stored in the index.

3.3 Information Extraction

From indexes of artist-related Web pages, a multitude of information can be derived. Extracting the different pieces of information according to the categories described in Chapter 2 largely depends on the textual representation of the Web pages. Even the information categories, for which this is not obvious at first glance, e.g., artist similarities, album cover artwork, or multimedia content, are derived from textual data.

Most approaches to text-based information retrieval rely on some principal assumptions and models, which are detailed in the following, before the particular information extraction techniques employed in the context of this PhD thesis are presented. The *bag of words* model, which can be traced back at least to [Luhn, 1957], represents a document as an unordered set of its words, ignoring structure and grammar rules. Words can be generalized to terms, where a *term* may be a single word or a sequence of n words (*n-grams*), or correspond to some grammatical structure, like a noun phrase. Using such a bag of words representation, each term t describing a particular document d is commonly assigned a weight $w_{t,d}$ that estimates, e.g., the frequency or importance of t in/for d. Each document can then be described by a *feature vector* comprising the single weights. When considering a whole corpus of documents, each document can be thought of as a representation of its feature vector in a *feature space* or *vector space* whose dimensions correspond to the particular term weights. This so-called *vector space model* is a fundamental model in information retrieval and was originally described in [Salton et al., 1975].

When it comes to deriving artist-related information from the Web, usually all Web pages returned for a particular artist are regarded as one large, virtual document describing the artist under consideration. This aggregation seems reasonable since, in Web-based MIR, the usual entity of interest is the music artist, not a single Web page. Furthermore, it is easier to cope with very small, or even empty, pages if they are part of a larger virtual document.

[4] The following HTML attributes were used in the experiments: *href*, *src*, *value*, *data*, and *link*

3.3.1 Co-Occurrence Analysis

The analysis of term co-occurrences on artist-related Web pages is probably one of the most primitive approaches to derive artist-related information from the Web. The core idea is to count how often artist names are mentioned together on the same Web page. These counts allow for deriving artist similarities as well as measuring the prototypicality of an artist for a certain genre. The author proposes two different strategies for performing co-occurrence analysis. The first one is based on *page counts* returned by a queried search engine, while the second one analyzes the *content of the top-ranked pages* returned by the search engine. Both strategies take as input a list of artist names.

Page Counts

Employing the page counts strategy, a Web search engine is used to estimate the number of Web pages containing each artist name and each pair of artist names. For this purpose, queries of the form "*artist name i*" and "*artist name i*"+"*artist name j*" are issued, respectively. Since this approach ignores the content of the found Web pages, Web traffic can be minimized by restricting the search to display only the top-ranked page if the queried search engine offers such an option. This obviously also raises performance. Subsequently, the page counts are used to create a probabilistic model, from which artist similarities and prototypicalities can be derived. To this end, the page counts are represented by a symmetric *co-occurrence matrix* C, where element c_{ij} gives the number of Web pages that mention both the artist name with index i and the one indexed by j. The values of the diagonal elements c_{ii} show the total number of pages mentioning the name of artist i.

Based on the page counts matrix C, relative frequencies are used to compute a *conditional probability matrix* P as follows. Given two events E_i (artist with index i is mentioned on a particular Web page) and E_j (artist with index j is mentioned on the particular Web page), the conditional probability p_{ij}, i.e., the probability for artist j to be found on a Web page that is known to contain artist i, is estimated as shown in Formula 3.1.

$$p\left(E_i \wedge E_j \mid E_i\right) = \frac{c_{ij}}{c_{ii}} \tag{3.1}$$

P thus gives a similarity matrix, which is obviously not symmetric. Of course, P can easily be symmetrized by calculating, e.g., the arithmetic mean of p_{ij} and p_{ji} for all pairs (i, j). In such a symmetric version, P may be used, for example, for classifying new artists into a given genre taxonomy, e.g., [Schedl et al., 2005a] or for generating playlists with similar pieces of music, e.g., [Knees et al., 2006a], [Logan, 2002]. However, the asymmetry information in P can also be used beneficially to detect prototypical artists, as described below.

To summarize, the page counts approach offers the advantages of requiring only very little disk space

and reducing Web traffic to a minimum. On the other hand, this approach scales poorly as the number of queries that has to be issued to the search engine grows quadratically with the number of artists in the collection. This quadratic computational complexity can be avoided when employing the second strategy to co-occurrence analysis, which will be presented in the following.

Content Analysis of Top-Ranked Web Pages

An alternative to the simple page counts approach is to retrieve, for each artist i, a certain amount of top-ranked Web pages[5], according to the search engine's judgment. After having performed indexing, cf. Subsection 3.2, the pages returned for artist i are searched for occurrences of artist name j, and the number of these occurrences are again used to fill a co-occurrence matrix C. When pursuing this strategy, element c_{ij} gives the number of Web pages retrieved for artist i that also mention artist j. This number, in fact, represents a document frequency, cf. Subsection 3.3.3. The diagonal elements c_{ii} reveal the total number of Web pages retrieved for artist i. This number does not necessarily correspond to the number of pages returned by the search engine for artist i, since usually between 5% and 20% of the pages are irretrievable due to various reasons, e.g., broken links or temporary server errors. Employing this approach, C will usually be asymmetric since, for a given pair of artists (i, j), the number of occurrences of j's name on Web pages known to contain i's name will differ from the number of i's occurrences on j's Web pages. A conditional probability matrix P is then derived exactly as described in the previous subsection. Compared to the page counts approach, the content analysis requires sending much less queries to the used search engine. On the other hand, retrieving and indexing the top-ranked pages also take time. As one is usually restricted by limitations on the number of queries allowed to issue per day, which are imposed by major providers of search engines, the simple page counts approach is barely feasible for collections exceeding some tens or hundreds of artists. Hence, content analysis of top-ranked pages is preferable in most real world applications.

3.3.2 Backlink/Forward Link Analysis

Based on the asymmetric conditional probability matrix P, as introduced in the last subsection, it is possible to estimate the prototypicality of an artist for a genre (or any other aggregation of artists that share some property). The author of this PhD thesis regards the prototypicality of a music artist as being strongly related to how often music-related Web pages refer to the artist and builds a model upon this consideration. The model's basic assumption is that phrases like "X sounds like Y" are more likely to occur on Web pages of a rather unknown band or artist X, indicating a similarity to a popular

[5] For the experiments conducted and the creation of AGMIS, retrieving at most 100 Web pages per artist seemed sufficient.

band or artist Y. For example, it is more natural to say that the Finnish heavy metal band *Sentenced* sounds like the well known pioneers *Metallica* than vice versa. This is due to the fact that *Metallica* is a more prototypical artist for the genre heavy metal than *Sentenced*.

The proposed approach is based on an idea similar to the *PageRank citation ranking* [Page et al., 1998] used by Google. Page et al. define a *forward link* of a Web page w as a link that is placed on w and links to another Web page. A *backlink* of a Web page w, in contrast, is defined as a link on any Web page other than w that links to w. Since the author's approach is based on investigating co-occurrences rather than links, the above definitions need slight modifications. In the proposed prototypicality estimation model, the number of backlinks of an artist of interest i is calculated by focusing i and counting how many Web pages that are known to mention another artist also mention artist i. Thus, any co-occurrence of artists i and j (unequal to i) on a Web page that is known to mention artist j is called a *backlink* of i from j. A *forward link* of an artist of interest i to another artist j, in contrast, is given by any occurrence of artist j on a Web page that is known to mention artist i. Using this interpretation of a backlink and a forward link, the author proposes a model for artist prototypicality ranking, which is introduced in the following and extended in the subsequent subsection.

Backlink/Forward Link Ratio

To obtain an estimation of the prototypicality of an artist i for a genre g, for each pair of artists $(i,j)_{j\neq i}$, it is investigated whether the number of backlinks of i from j exceeds the number of forward links of i to j. Recording for how many of the artists j from the same genre as i this is the case yields counts for backlinks (and forward links) on the artist level. The larger this backlink count, the higher the probability for artist i being mentioned in the context of other artists from the same genre g and thus, the higher the prototypicality of i for g.

Formally, using the similarity matrix P, which results from Formula 3.1, the ranking function $r(i,g)$ that describes the prototypicality of artist i for genre g is given by Formula 3.2, where n is the total number of artists in g and $bl(i,j)$ and $fl(i,j)$ are functions that return a boolean value according to Formulas 3.3 and 3.4, respectively.

$$r(i,g) = \frac{\sum_{j=1}^{n,j\neq i} bl(i,j)}{\sum_{j=1}^{n,j\neq i} fl(i,j)} \tag{3.2}$$

$$bl(i,j) = \begin{cases} 1 & \text{if } \frac{c_{ij}}{c_{ii}} < \frac{c_{ji}}{c_{jj}} \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

$$fl(i,j) = \begin{cases} 1 & \text{if } \frac{c_{ij}}{c_{ii}} \geq \frac{c_{ji}}{c_{jj}} \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

Hence, $bl(i,j)$ gives the value 1 if artist i has more backlinks from artist j (relative to the total number of Web pages retrieved for artist j) than forward links to artist j (relative to the total number of Web pages retrieved artist i). $fl(i,j)$ is defined analogously. The author calls $r(i,g)$ the *backlink/forward link (BL/FL) ratio* of artist i since it counts how often the relative frequency of backlinks of i exceeds the relative frequency of its forward links and relates these two counts. Performing the above calculations for all pairs of disjoint artists (i,j) from a genre g eventually yields a complete artist prototypicality ranking for g.

Backlink/Forward Link Ratio with Penalization of Exorbitant Popularity

A drawback of the BL/FL model is that artist names that equal common speech words are always top-ranked.[6] This is due to the fact that such words frequently occur on arbitrary Web pages, without referring to the artist, but to the common speech word. Therefore, they create a lot of unjustified backlinks for artists with the respective names and hence distort the prototypicality ranking.

To reduce such distortions, the author proposes a refinement to the BL/FL model, which basically adopts the idea of the popular term weighting approach TF·IDF, cf. Subsection 3.3.3, by awarding high intra-genre BL/FL ratios, but penalizing the prototypicality of an artist if it is high over all genres. This is reasonable since even very popular and important artists are unlikely to be prototypes for all genres. Hence, the extended model penalizes artists with exorbitant popularity over all genres. In accordance with the naming scheme of TF·IDF, this approach is called GP·IOP for *genre prototypicality · inverse overall prototypicality*.

Incorporating information about the overall prototypicality of the artist under consideration, the ranking function $r(i,g)$ of the extended model is given in Formula 3.5, where n is the number of artists in genre g. The penalization term, i.e., the IOP factor, is given in Formula 3.6, where m is the total number of artists in the collection. The functions $bl(i,j)$ and $fl(i,j)$ are defined as in Formulas 3.3 and 3.4, respectively. The normalization function $\|\cdot\|$ shifts all values to the positive range and maps them to $[0,1]$.

$$r(i,g) = \frac{\sum_{j=1}^{n,j\neq i} bl(i,j)}{\sum_{j=1}^{n,j\neq i} fl(i,j) + 1} \cdot penalty(i)^2 \tag{3.5}$$

$$penalty(i) = \left\| \log \left(\frac{\sum_{j=1}^{m,j\neq i} fl(i,j) + 1}{\sum_{j=1}^{m,j\neq i} bl(i,j) + 1} \right) \right\| \tag{3.6}$$

[6] For the 224-artist-collection used in [Schedl et al., 2005b], the highest overall BL/FL ratios were obtained for the artists *Bush* (223/0), *Prince* (222/1), *Kiss* (221/2), *Madonna* (220/3), and *Nirvana* (218/5).

3.3.3 Term Weighting

From the bag of words representation of a set of retrieved and indexed Web pages, as introduced in the beginning of this section, a *relevance score* of a term t for a single document d, or for a virtual, aggregated document describing a particular artist a, can be calculated. The general idea of assigning a weight to "notions" in a document can be traced back to [Luhn, 1957]. In modern information retrieval, this idea was formalized, e.g., in [van Rijsbergen, 1979], [Salton and Buckley, 1988], and [Robertson and Jones, 1988]. There exists a large variety of different formulations of such a relevance score, which can be regarded as a term weight $w_{t,d}$ for a document d or $w_{t,a}$ for a virtual document describing an artist a. An overview of some frequently used formulations is given, for example, in [Zobel and Moffat, 1998] and [Salton and Buckley, 1988]. Reviewing the literature, the authors of [Zobel and Moffat, 2006] distill the following three principal *monotonicity assumptions* that underly most term weighting approaches.

1. Less weight is given to terms that appear in many documents.
2. More weight is given to terms that appear many times in a document.
3. Less weight is given to documents that contain many terms.

These principal assumptions aim at emphasizing terms that are discriminative for a document, while reducing the weight of terms that are less informative. The intention of the first assumption, sometimes also called the *IDF (inverse document frequency) assumption*, is that terms appearing in many documents bear less information than terms appearing only in a small number of documents. To give an example, in the context of MIR, a term like "music" will probably not be very discriminative for a particular artist and will thus have a low IDF value. The second assumption, also called the *TF (term frequency) assumption*, ensures that a term is given a high weight if it occurs frequently in a document, which seems reasonable per se. In the context of MIR, terms with high term frequencies are, for example, album names on Web pages containing reviews of new album releases or rather general terms like "song" or "band". The intention of the third assumption is to *normalize* a document's feature vector with respect to the document length. The appearance of a particular term in a long document should be less important than its appearance in a short document. Normalization with respect to document length is usually addressed by calculating the *cosine similarity* between the feature vector representations of two documents. The cosine similarity equals the cosine between the angle of the feature vectors, which are represented as unit vectors. The first reference to the cosine similarity in this context is probably [Salton, 1962].

The first and second monotonicity assumptions are usually addressed by a formulation of the term

weighting function that is commonly known as TF·IDF (term frequency · inverse document frequency). In the context of this PhD thesis, the TF·IDF formulation given in Formula 3.7 is used as it proved to yield good results for clustering music artists – cf. [Knees et al., 2004]. In Formula 3.7, n is the total number of Web pages retrieved for all artists in the collection, $tf_{t,a}$ is the number of occurrences of term t in the virtual document of artist a, and df_t is the number of documents in which t occurs at least once. The third monotonicity assumption is addressed by calculating the cosine similarity between the term weight vectors of each pair of artists (a, b) according to Formula 3.8, where $|T|$ is the cardinality of the term set, i.e., the dimensionality of the term weight vectors. In this formula, θ gives the angle between a's and b's feature vectors in the Euclidean space.

$$w_{t,a} = \begin{cases} (1 + \log_2 tf_{t,a}) \cdot \log_2 \frac{n}{df_t} & \text{if } tf_{t,a} > 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.7}$$

$$sim(a, b) = \cos\theta = \left(\frac{\sum_{t=1}^{|T|} w_{t,a} \cdot w_{t,b}}{\sqrt{\sum_{t=1}^{|T|} w_{t,a}^2} \cdot \sqrt{\sum_{t=1}^{|T|} w_{t,b}^2}} \right) \tag{3.8}$$

Term weighting functions are employed in this thesis to determine a set of descriptive terms that are most relevant to an artist of interest and to calculate similarities between artists.

3.3.4 Natural Language Processing for Determining Band Members

The task of determining the members of a music band and the instruments they play, cf. Subsection 2.2.3, could, in principal, be addressed by co-occurrence analysis, as introduced in Subsection 3.3.1. However, even frequent occurrences of an individual artist a somewhere on a Web page retrieved for a band b do not necessarily give strong indication for a being a member of b. Instead, it is highly probable that such occurrences are due to the fact that a is compared with b's respective member, e.g., "a would be a good drummer for b", especially if a is very popular. Such occurrences can also mean that a played a concert together with b, without being a member of b. Therefore, methods that perform a deeper content analysis than just scanning a Web page for occurrences of particular terms are needed to tackle this task.

The approach presented here includes three steps based on natural language processing, which are elaborated on in the following subsections. First, possible band members are sought by employing a *named entity detection* (NED) technique. A good outline of the evolution of NED can be found, e.g., in [Callan and Mitamura, 2002]. The second step is to search for specific linguistic patterns in the text surrounding the potential members. According to the frequency of these patterns, (member,

vocals singer	vocal, voice, voices vocalist, voice, chanter, chanteuse, choralist, chorister, crooner, minstrel
guitar guitarist	guitars
bass bassist	bass guitar bass guitarist
drum drummer	drums, percussion, percussions percussionist
keyboard keyboardist	keyboards, key-board, key-boards keyboarder, keyboard player

Table 3.1: Synonyms for instruments and roles used in band member detection.

instrument)-pairs for the band under consideration are eventually predicted in a third step.

From a methodological point of view, the work presented in [Cimiano et al., 2004] relates to the work presented in this PhD thesis in regard to the task of determining band members. In fact, Cimiano et al. propose a pattern-based approach to finding instances of concepts on Web pages and classify them according to an ontology of concepts. To this end, the page counts returned by Google for search queries containing hypothesis phrases are used to assign instances to concepts. For the general geographic concepts, like city, country, or river, and the well known instances used in the experiments, this method yielded quite promising results. In contrast, the task of assigning (member, instrument)-pairs to music bands is a more specific one. The author conducted preliminary experiments on using the page counts returned for patterns including instrument, member, and band names, but results were very poor. Querying such patterns as exact phrases, the number of found Web pages is very small, even for well known bands and members. Using conjunctive queries instead does not work either as the results are, in this case, heavily distorted by famous band members frequently occurring on the Web pages of other bands. For example, *James Hetfield*, singer and rhythm guitarist of the band *Metallica*, occurs in the context of many other heavy metal bands. Thus, he would likely be predicted as the singer (or guitarist) of a large number of bands other than *Metallica*.

In the context of this PhD thesis, instrument detection is restricted to the standard line-up of most rock bands, i.e., to singer(s), guitarist(s), bassist(s), drummer(s), and keyboardist(s) for performance reasons. The approach presented in the following can be generalized, however, to reveal arbitrary roles within a band.

Named Entity Detection

In a first step, named entities that may be band members are sought in the Web pages retrieved for the band under consideration by *analyzing word capitalization* and *filtering*. Assuming the name of a band member to comprise at least two and at most four single names, which holds for the vast

majority of artists, *2-*, *3-*, and *4-grams* are extracted from the plain text representation of the Web pages. Subsequently, some basic filtering is performed. Those n-grams whose substrings contain only one character are discarded, and only those n-grams whose tokens all have their first letter in upper case and all remaining letters in lower case are retained. Finally, using the *iSpell English Word Lists* [isp, 2006], all n-grams containing at least one substring that is a common speech word are filtered out. The remaining n-grams are regarded as potential band members.

Rule-Based Linguistic Pattern Analysis

Having determined the potential band members via NED, a linguistic analysis step is performed to obtain the actual instrument(s) of each member. Similar to the approach proposed in [Hearst, 1992] for finding hyponyms in large text corpora, the author defined the following rules and applied them on the potential band members and the surrounding text as necessary.

1. *M* plays the *I*
2. *M* who plays the *I*
3. *R M*
4. *M* is the *R*
5. *M*, the *R*
6. *M* (*I*)
7. *M* (*R*)

In these rules, *M* is the potential band member, *I* is the instrument, and *R* is the role *M* plays within the band (singer, guitarist, bassist, drummer, keyboardist). To cope with the use of different words for the same concept, *synonym lists* gathered from *Thesaurus.com* [the, 2007] and from the online dictionary *LEO* [leo, 2007] are used for *I* and *R*. The synonyms are listed in Table 3.1. It is then counted on how many of the Web pages each rule applies for each *M* and *I* (or *R*). These counts are summed up for all synonyms of each concept, yielding (member, instrument, rule, count)-quadruples.

Prediction According to Document Frequencies

Since the aggregated counts indicate, for example, that on 24 of the Web pages returned for the search query `"Primal Fear"+music` *Ralf Scheepers* is said to be the singer of the band according to rule 6, on 6 pages according to rule 3, and so on, these counts are, in fact, document frequencies. They

are stored as a set of (member, instrument, rule, DF)-quadruples for every band. Subsequently, the document frequencies given by the individual rules are summed up over all (member, instrument)-pairs of the band under consideration, which yields (member, instrument, ΣDF)-triples. To reduce uncertain membership predictions, triples whose ΣDF values are below a threshold t_{DF}, expressed as a fraction of the highest ΣDF value of the band under consideration, are filtered out. To give an example, this filtering would exclude, in a case where the top-ranked singer of a band achieves an accumulated ΣDF value of 20, but no potential drummer scores higher than 1, all potential drummers for any $t_{DF} > 0.05$. Thus, the filtering would discard information about drummers since predictions based on only one rule appliance on one Web page may be faulty with high probability.
Eventually, all (member, instrument)-pairs that remain after the filtering described above are predicted, which enables an `m:n` assignment between instruments and members.

3.3.5 Image Retrieval

Considering the trend towards digital music distribution and the resulting absence of any physical manifestation of albums when purchasing music in this vein, digital images of album cover artwork are becoming an important category of music-related metadata. Subsection 2.3.2 already showed some sources of such digital cover artwork. Automatic retrieval of album cover images, given only the names of album and artist under consideration, may be performed by relying on common image search engines provided by major Web search providers. However, this approach requires to select the correct album cover image from all results returned by the search engine. Always choosing the top-ranked image may be a solution, though generally not a very good one. In fact, preliminary experiments conducted by the author showed that the top-ranked image returned by Google's image search engine to queries of the form "*artist name*"+"*album name*"+cover is often incorrect, but sometimes followed by a number of correct album cover images.

Character and Tag Distance

An alternative to the use of specialized image search engines is to create a full inverted index including the plain text and the HTML tags of the retrieved artist-related Web pages. Using the information on term positions, hereafter, the text included in the <img> tags and the text surrounding them is analyzed for album and artist names. This idea is similar to the approach proposed in [Tsymbalenko and Munson, 2001], where the authors found that information encoded in an image's file name (stored in the *src* attribute of the <img> tag), in the Web page's title, and in the *alt* attribute of the <img> tag provides important clues for an image's content. Generalizing these find-

ings, the author of this PhD thesis proposes the use of *character and term distances* between <img> tags referencing potential album cover images and names of artist or album under consideration to estimate the probability of an image to depict an album cover. This generalization obviously also assigns a high relevance to occurrences of album or artist name in the *src* or *alt* attribute of the <img> tag under evaluation. Determining a fixed number of images with the smallest sum of the distances $|<img>\ tag - artist\ name|$ and $|<img>\ tag - album\ name|$ yields a set of potential album cover images.

Content-Based Filtering

After having determined a set of potential cover images, e.g., as a result of queries to an image search engine or of distance measurement between <img> tags and textual identifiers, the author proposes to apply some content-based filtering techniques to eliminate erroneous images. Taking the almost quadratic shape of most album covers into account, all potential cover images that have non-quadratic dimensions within a tolerance of 15% are rejected. Applying this simple constraint, an improvement in accuracy by more than 3 percentage points in average is achievable – cf. Section 5.3.5. While this filtering remedies problems with misdimensioned images, it cannot distinguish between actual covers and scanned discs, which are also often present in the set of potential cover images. To address this issue, the author proposes a simple circle detection technique. Usually, such images of scanned discs are cropped to the circle-shaped border of the compact disc, which allows to use a simple algorithm instead of complex circle detection techniques. To this end, small rectangular regions along a circular path that is touched by the image borders tangentially are examined, and the contrast between subareas of these regions is determined using *color histograms*, e.g., [Feng et al., 2003].[7] Since images of scanned compact discs show a strong contrast between subareas showing the imprint and subareas showing the background, the pixel distributions in the highest color value bins of the histograms are accumulated for either type of region (imprint and background). If the number of pixels in the accumulated imprint bins exceeds or falls short of the number of pixels in the accumulated background bins by more than a factor of 10, this gives strong evidence that the image under evaluation shows a scanned disc. In this case, the respective image is discarded.

Image Selection

After the content-based filtering step, the actual prediction of one image that most likely shows the album cover is finally made either by selecting the image whose reference in the Web page shows

[7] The RGB color space is used for histogram generation.

minimal character or tag distance to textual identifiers, cf. Subsection 3.3.5, or by deriving an average, normalized color histogram from the set of potential cover images and choosing the image whose normalized histogram is most similar to this average histogram.

Chapter 4

Techniques Related to Information Visualization

As the field of information visualization (InfoVis) is widespread, this chapter will focus on methods that have been developed in the context of this PhD thesis and on related work. More precisely, Section 4.1 presents a graph-based approach to visualize artist prototypicalities for genres, which has been developed by the author of this thesis. To embed this approach in a larger context, an overview of similar visualization techniques for other MIR-related purposes, especially for illustrating similarity relations, is given beforehand. To stay within the scope, the presentation of related approaches will focus on graph-based techniques.

Section 4.2 focuses on InfoVis approaches to illustrate hierarchically structured data. Besides well established techniques, like *Treemap* and *Sunburst* visualizations, a novel method proposed by the author of this thesis and called *Stacked Three-Dimensional Sunbursts* is presented. A user interface that employs a variant of this novel technique is integrated in AGMIS, cf. Chapter 6. Details on this user interface, which is called *Co-Occurrence Browser* (COB), are given in Subsection 4.2.6. In short, COB automatically structures a sets of Web pages in a hierarchical manner according to co-occurring terms. Furthermore, a second tree visualization technique developed by the author, the *Circled Fans*, is briefly presented and related to other techniques.

4.1 Visualization Methods for Relations Between Artists

In MIR probably the most important type of relationship between artists or between songs is similarity relations, cf. Subsection 2.1.1. Consequently, there exists a wide variety of visualization methods to illustrate such similarity relations. Since similarity data for a particular music entity can be seen as a representation of the music entity in a high-dimensional feature space, visualizing such data is strongly related to data projection as the available visualization space is usually two-dimensional. The same holds for other representations of music entities in a feature space, e.g., given by the output of a particular signal-based or context-based feature extraction algorithm.

4.1.1 Self-Organizing Maps

A large group of visualization techniques is formed by the various approaches building upon the *Self-Organizing Map* (SOM) [Kohonen, 1982], [Kohonen, 2001]. The SOM is a non-linear data projection method to map feature vectors, e.g., representations of pieces of music or of music artists, from a high-dimensional feature space to a usually two-dimensional output space. The output space is a discrete space, composed of a set of *map units*. After having trained the SOM, which can also be seen as a neural network for unsupervised learning, each data item can be assigned a particular map unit that best represents it – its so-called *best matching unit*. Like any other reasonable data projection method, the SOM aims at topology preservation, i.e., items that are close to each other in the feature space should be mapped to close map units in the output space.

An overview of SOM-based data visualization methods is given in [Vesanto, 1999] and [Vesanto, 2002]. Among the most popular ones is the *U-matrix* [Ultsch and Siemon, 1990], which illustrates by color the distances between neighboring map units of the SOM. In [Pölzlbauer et al., 2005a] Pölzlbauer et al. present the *gradient field* method, which is similar to the U-matrix, but encodes similarity information by drawing a vector field on top of the SOM lattice, where each arrow points to its nearest cluster center. *Component planes* are also quite popular as they allow to illustrate the distribution of the values of each data dimension over the SOM. The *Smoothed Data Histogram* (SDH) [Pampalk et al., 2002b] reveals the clustering by first letting vote each data item for a fixed number of map units that best represent it, second smoothing the resulting voting matrix via interpolation, and third depicting this smoothed version on top of the SOM lattice. Variants of the SDH have been successfully applied to visualize similarities between music entities, e.g., in [Pampalk et al., 2004], [Schedl, 2003], and [Knees et al., 2006b].

A graph-based approach to visualize similarity relations on top of a SOM lattice is presented in [Pölzlbauer et al., 2005b]. Pölzlbauer et al. propose two methods to create a graph in the high-dimensional feature space, i.e., the input space of the SOM, and visualize this graph in the low-dimensional projection space as an overlay on the SOM grid. A graph is formally defined as a tuple $\langle V, E \rangle$, where V is a set of nodes (vertices) and E is a set of edges, each of which connects exactly two disjoint elements from V. V is given by the data items. The authors define E using either a fixed number k of each data item's *nearest neighbors* in the feature space or, alternatively, by using a threshold for the minimum distance two data items must have in order to create an edge between their corresponding vertices. Applying the first method to determine E hence inserts, for each node, an edge to its k nearest neighbors. Following the second approach, which is called *radius method* by the authors, each pair of nodes is considered, and an edge between two nodes is inserted into E iff the distance between their corresponding data items is below a fixed threshold. After having created

the graph in the feature space, it is projected into the output space, i.e., the usually two-dimensional projection space of the SOM. This is performed in a straightforward manner by interpreting each map unit of the SOM as a node and mapping each element v_i from V to the node representing v_i's best matching unit. E is preserved in such a way that connected data items in the original graph translate to connected map units (the respective best matching units for the two data items under consideration). Drawing the projected graph on the SOM lattice reveals which regions of the SOM are densely populated and which are not. Moreover, this visualization technique supports locating interpolating map units and outliers.

4.1.2 Multi-Dimensional Scaling

Multi-Dimensional Scaling (MDS), e.g., [Kruskal and Wish, 1978], [Cox and Cox, 1994], refers to a category of approaches to non-linear data projection for input data in the form of a distance matrix. The basic idea is to position the data items in the low-dimensional projection space such that the original distances between the data items in the feature space are approximated by the corresponding distances in the projection space. The common algorithms initially assign a random position to each data item in the projection space or use some kind of linear projection, e.g., *Principal Components Analysis* [Hotelling, 1933], to obtain an initial mapping. Subsequently, the data items are repositioned in the projection space, trying to minimize an *error function*, which is the main objective of MDS. The error function is sometimes also called *stress*, *strain*, *loss*, or simply *costs*, depending on the interpretation. Denoting the $n \times n$ distance matrix given as input as D, where d_{ij} gives the distance between data items x_i and x_j, and denoting the distance between x_i and x_j in the projection space as d'_{ij}, there exists a variety of different formulations of the error function. Usually, distances are calculated in the Euclidean metric, although, in principal, any distance measure (metric or non-metric) that seems appropriate for the data under consideration can be used. A simple choice for the error function is the sum of the squared pairwise errors, as shown in Equation 4.1.

$$E = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(d_{ij} - d'_{ij}\right)^2 \tag{4.1}$$

Another frequently used error function was proposed by Sammon in [Sammon, 1969], whose approach to MDS was later named after him as *Sammon's mapping*. Sammon proposed the error function given in Equation 4.2. This function normalizes each pairwise error by the distance of the respective data items in the input space. Due to this normalization, Sammon's mapping emphasizes the preservation

of small distances.

$$E = \frac{1}{\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} d_{ij}} \cdot \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \frac{\left(d_{ij} - d'_{ij}\right)^2}{d_{ij}} \tag{4.2}$$

In the context of MIR, MDS has been applied, for example, in [Cano et al., 2002], where the authors use an MDS variant optimized for computational efficiency to arrange songs on a plane according to their similarities. The input data is given as Euclidean distances between low-level signal-based feature representations of the songs under consideration.

Another work related to MIR that makes use of MDS can be found in [Seyerlehner, 2006], where Seyerlehner first extracts high-level audio features from an arbitrary collection of songs to subsequently model a *k-nearest neighbor graph* (k-NNG) [Eppstein et al., 1997] using distances calculated from the audio features. On the k-NNG representation of the music collection, MDS is applied to lay out the high-dimensional data in the two-dimensional visualization space. The calculatory restriction to a subset of the complete distance matrix, which is imposed by the k-NNG, allows for a performant computation of the MDS. Subsequently, each data item is represented by a two-dimensional Gaussian distribution whose mean μ equals the item's position on the visualization plane and whose variance σ^2 is set to a constant value for all items. Finally, all data items' Gaussian representations are aggregated and yield a visualization, which is integrated into a user interface for browsing music collection, called the *Music Browser*. A screenshot of the Music Browser is depicted in Figure 4.1.

4.1.3 Similarity Networks

Graph-based visualization approaches, also called *network visualizations* or *node-link diagrams*, are scarcely used to illustrate relationships between music entities. Besides the Music Browser, which was already addressed in the previous subsection, Cano and Koppenberger in [Cano and Koppenberger, 2004] model similarity relations between artists as a similarity network and analyze some of its properties. In particular, they construct artist similarity graphs from two sources: expert opinions on similar artists extracted from AMG and playlist co-occurrences. More details can be found in Subsection 2.1.1 and in [Cano and Koppenberger, 2004].

Another graph-based model is used in [Vignoli et al., 2004] to visualize an artist similarity matrix in a two-dimensional space. Each artist is represented by a vertex, and an edge between two vertices is drawn if the corresponding artist similarity is greater than a certain threshold. To position the vertices in a visually appealing manner, Vignoli et al. use a physical spring model, where vertices are interpreted as metal rings and edges as springs. The respective distances are interpreted as repelling or attracting forces. The aim is then to bring the system in a state of low energy by minimizing an

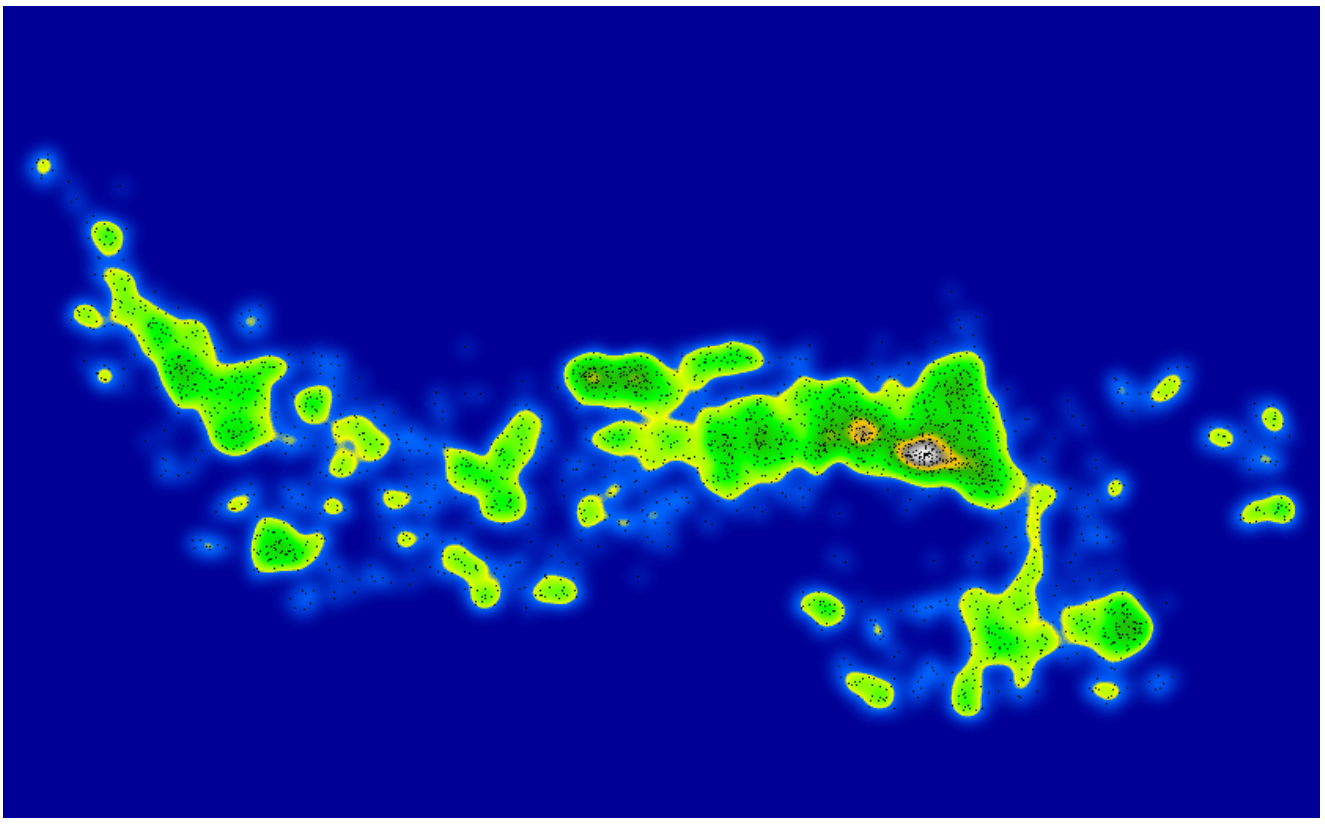


Figure 4.1: Music Browser visualization using Multi-Dimensional Scaling to organize songs.

energy function. In comparison with MDS, this so-called *Spring Embedder* approach, cf. [Eades, 1984], [Fruchterman and Reingold, 1991], only calculates a layout on the low-dimensional visualization space and is thus not suited for high-dimensional data projection. To give a meaning to certain regions of the visualization plane, Vignoli et al. extend the Spring Embedder model by inserting attractors, which can be thought of as magnets. These attractors represent certain musical properties, like the ones listed in Subsection 2.2.1. The definition of the energy function is modified accordingly by a term that represents a measure of relevance of each property for each artist. The resulting visualization is finally embedded into a user interface to browse music collections on mobile devices.

A graph-based visualization technique called *Probabilistic Network*, which employs a variant of MDS for data projection, has also been integrated by the author of this PhD thesis into the CoMIRVA framework. Given a similarity matrix as input, the Probabilistic Network algorithm first places the data items on random positions in the two-dimensional visualization space. To reorganize the positions of the artists in the output space, a *simulated annealing* approach [Kirkpatrick et al., 1983], [van Laarhoven and Aarts, 1987] is used. To this end, an iterative algorithm randomly selects two nodes and adjusts the distance between them in the output space to better fit the corresponding distance in the high-dimensional feature space. A time-dependent, linearly decreasing rate of adaptation is further used to gradually decrease the maximum distance correction. The reason for this is that high

adaptations are necessary in the beginning due to the random initial placement of the nodes, whereas lower adaptations towards the end of the algorithm are intended to stabilize the system and should therefore only fine-tune the positions. The error function to be minimized is the function proposed in [Sammon, 1969], cf. Subsection 4.1.2. After having laid out the artists, the connectivity of the graph is determined by an approach similar to the Erdős Rényi random graph, cf. [Erdős and Rényi, 1959]. While the approach by Erdős and Rényi inserts an edge between each pair of vertices with a fixed probability p, the Probabilistic Network uses the $[0,1]$-normalized similarity matrix S $[s_{ij}]$ given as input and inserts an edge between two nodes x_i and x_j if, for a fixed parameter $prob\_corr$, the condition $s_{ij} > rand \cdot prob\_corr$ holds. In this condition, $rand$ is a random value in the range $[0,1]$. Figure 4.2 illustrates two Probabilistic Network visualizations using as input an artist similarity matrix created via co-occurrence analysis, cf. Subsection 3.3.1. From these screenshots, the impact of the parameter $prob\_corr$ on the appearance of the network can be seen. For the left network, $prob\_corr$ was set to 0.5 while for the right one, a value of 5.0 was used.

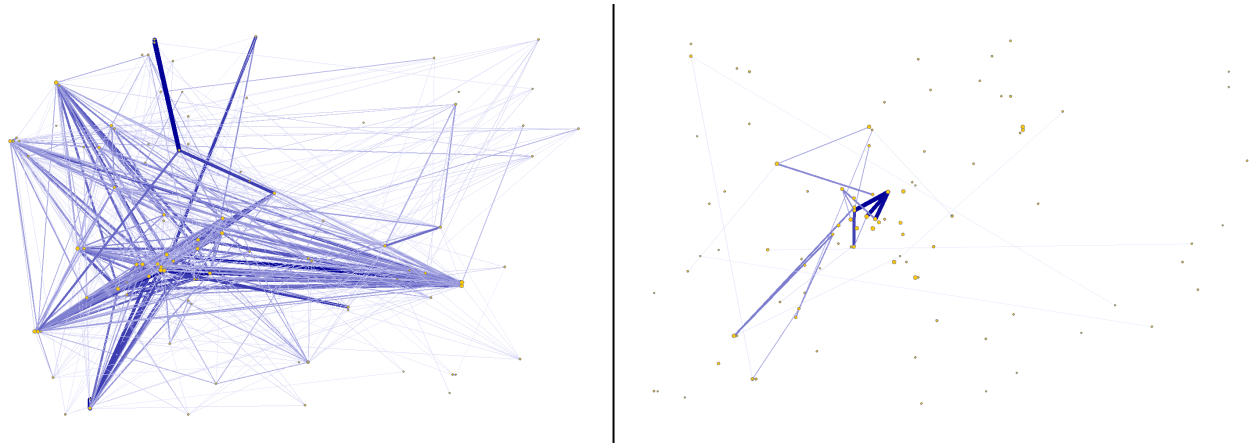


Figure 4.2: Two Probabilistic Network visualizations based on Multi-Dimensional Scaling and random graphs.

4.1.4 Continuous Similarity Ring

In order to visualize the prototypical artists for a music genre together with similar artists, as identified by the methods described in Subsections 3.3.1 and 3.3.2, the author of this PhD thesis developed a novel visualization technique called *Continuous Similarity Ring* (CSR). Details on the CSR are presented in [Schedl et al., 2005b] and summarized in the following.

The basic idea of the CSR is to organize the prototypical artists – one for each genre – in the form of a circle. Since similar or related prototypes and the genres they represent should be placed close to each other, a *Traveling Salesman Problem* (TSP), e.g., [Lawler et al., 1985] or [Skiena, 1997], is formulated, and a solution is sought using a simple heuristic algorithm. To this end, the asymmetric similarity matrix

S $[s_{ij}]$ obtained by co-occurrence analysis according to Subsection 3.3.1, is converted to a symmetric form S' $[s'_{ij}]$ by calculating the arithmetic mean of each pair s_{ij} and s_{ji}. The resulting matrix is transformed into a distance matrix D $[d_{ij}]$, which is fed into the TSP algorithm to find the shortest path between all prototypical artists. Ideally, the output is thus a tour that passes all prototypes and minimizes the overall distance. This tour defines the arrangement of the artists within the circle of prototypes.
Since the CSR also aims at showing which artists are similar to which prototypes, in a next step, the k nearest neighbors as given by S' are determined for each prototype p. These k neighbors are chosen from the complete artist set regardless of their assigned genre, which enables the user to easily make out artists that are inspired by musicians of different genres. Given the set of neighbors $N(p)$ for each prototype p, it is determined which artists are neighbor of only one prototype and which neighbor more than one prototype. The former ones are inserted into an artist set O, the latter ones into a set I. The goal is to point out artists that cannot be categorized into a particular genre and thus neighbor several prototypes. Each of these artists $n \in I$ is mapped to the area inside of the circle of prototypes and is connected to all prototypes p with $n \in N(p)$. The region outside of the circle of prototypes is used to display the elements in O.
Positioning the artists should be performed in a way that preserves the original distances between the artists and their prototypes as given by D. Furthermore, the length of the edges connecting prototypes and neighbors should be minimized in order to avoid overloading of the visualization. Thus, the CSR employs a heuristic cost-minimizing algorithm to position the artists in I. The costs $C(n)$ for an artist $n \in I$ are calculated as shown in Equation 4.3, where $P(n)$ is the set of prototypes that are connected to artist n, $d(p,n)$ is the original distance according to D between prototype p and neighbor n, and $d'(p,n)$ is the Euclidean distance on the visualization plane between the vertex representing prototype p and the vertex representing neighbor n. This cost function thus tries to preserve the proportion of the individual distances between artist n and each of n's prototypes to the sum of the distances between n and all of its prototypes.

$$C(n) = \sum_{p \in P(n)} \left(\frac{d(p,n)}{\sum_{q \in P(n)} d(q,n)} - \frac{d'(p,n)}{\sum_{q \in P(n)} d'(q,n)} \right) \tag{4.3}$$

The algorithm for positioning the vertex of a neighbor $n \in I$ comprises three steps, which are performed iteratively.

1. The vertex of the current neighbor $n \in I$ is initially positioned in the center of the screen.

2. n's position is then modified by a small, randomly chosen amount of up to 10 pixels.

3. The costs for the new position of n are calculated and the vertex is moved to the new position if an improvement in costs and in the term $\sum_{q \in P(n)} d'(q, n)$ (for minimizing the length of the edges) could be achieved.

Figure 4.3 shows a screenshot of a CSR visualization. The three nearest neighbors of each prototype are depicted, and edges connecting these neighbors with the respective prototypes are drawn. Varying thickness and color of the edges reveal similarity information about the corresponding artists. Thick and bright edges connect very similar artists, whereas thinner and darker edges connect artists with lower similarity values. Regarding Figure 4.3, it can be seen that the neighbors of the prototype *Johann Sebastian Bach* are not connected to any other prototype. Thus, it seems that classical artists are very well distinguishable from artists of other genres. It is also worth noting that *Willie Nelson* is one of the three nearest neighbors of *Johnny Cash* and of *Bob Dylan*, which nicely shows the overlap between the genres folk and country. Unfortunately, artists like *Bush*, whose names equal common speech words, are often found in the region inside of the circle of prototypes, having connections to unrelated artists. This is a problem of the underlying similarity measure, which is based on co-occurrences of artist names on Web pages. However, this problem arises only for small values of k. Using $k = 5$, for example, reveals more interesting relations, but at the cost of lucidity.

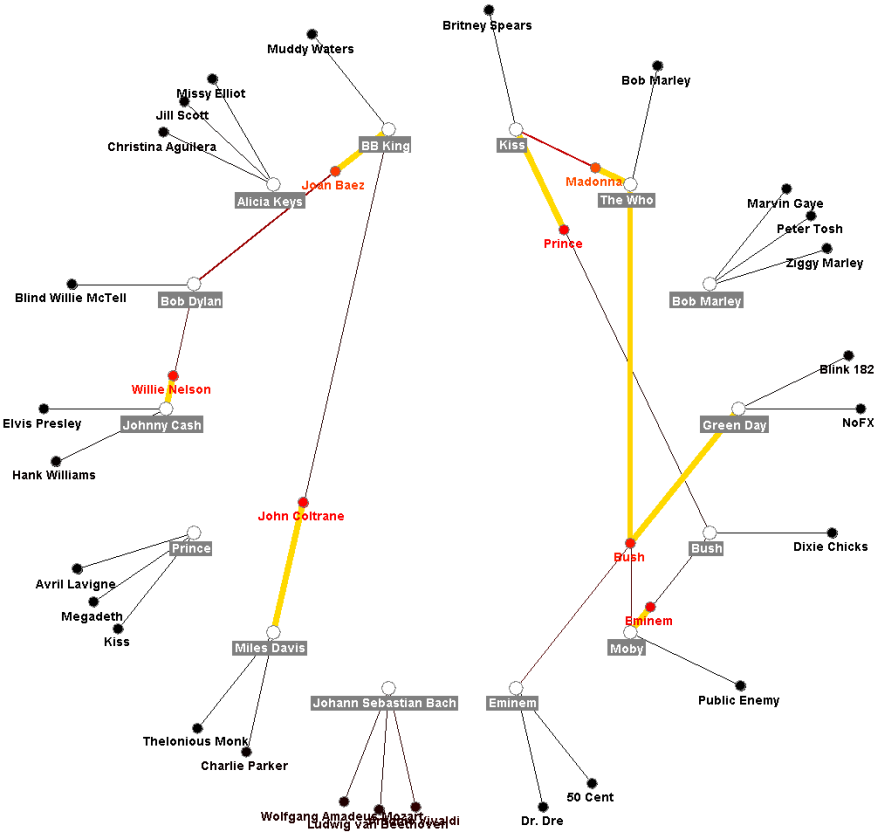


Figure 4.3: Continuous Similarity Ring visualization based on prototypical artists for 14 genres.

4.2 Visualization Methods for Hierarchical Data

In contrast to the representation of artist similarity data as not necessarily acyclic graphs, which was addressed in the last subsections, there exists a quite large amount of data that exhibits an intrinsic hierarchical structure. Such hierarchically structured data is usually represented by some kind of *tree*, i.e., a *connected acyclic graph*. As such tree structures occur in nearly every domain, considerable efforts to visualize them have been made. In the following subsections, an overview of the most popular visualization techniques for such hierarchical data is given. The discussion of these standard visualization approaches is complemented with elaborations on two methods developed by the author in the context of this thesis, namely the *Circled Fans* and the *Stacked Three-Dimensional Sunbursts*.

4.2.1 Treemap

A very popular space-filling visualization technique for hierarchically structured data is the *Treemap*, cf. [Johnson and Shneiderman, 1991], [Shneiderman, 1992]. The hierarchical structure of the underlying data is illustrated using a rectangular layout, which displays elements further down in the hierarchy embedded in the rectangle of their parent element. To this end, the largest rectangle, which represents the root node, is recursively partitioned into smaller regions according to some attribute of the data set. A sample screenshot of a Treemap visualization of a file system can be found in Figure 4.4. In this example, color is used to illustrate the different file types.

Since its initial proposal in 1991, many extensions of the Treemap have been elaborated. Two popular examples are *Cushion Treemaps* [van Wijk and van de Wetering, 1999] and *Squarified Treemaps* [Bruls et al., 2000]. While Cushion Treemaps were proposed to improve the perceptibility of the individual rectangles by adding an illumination effect via shading and spotlighting, Squarified Treemaps aim at subdividing the rectangular areas such that the resulting subrectangles have lower aspect ratios, therefore, using space more efficiently and simplifying estimating their size. Furthermore, a three-dimensional version of the Treemap, called the *StepTree*, is presented in [Bladh et al., 2004]. A user study of various Treemap-based applications and similar tree visualization systems can be found in [Kobsa, 2004].

4.2.2 Hyperbolic Browser / Hyperbolic Tree

Inspired by M. C. Escher's woodcut series *Circle Limit*, Lamping et al. propose in [Lamping et al., 1995] and [Lamping and Rao, 1996] the *Hyperbolic Browser*, which is sometimes also referred to as *Hyperbolic Tree* due to its underlying tree-based data structure and its tree-like appearance. The Hyperbolic Browser makes use of non-Euclidean geometry, e.g., [Coxeter, 1998], to visualize hierarchical data.

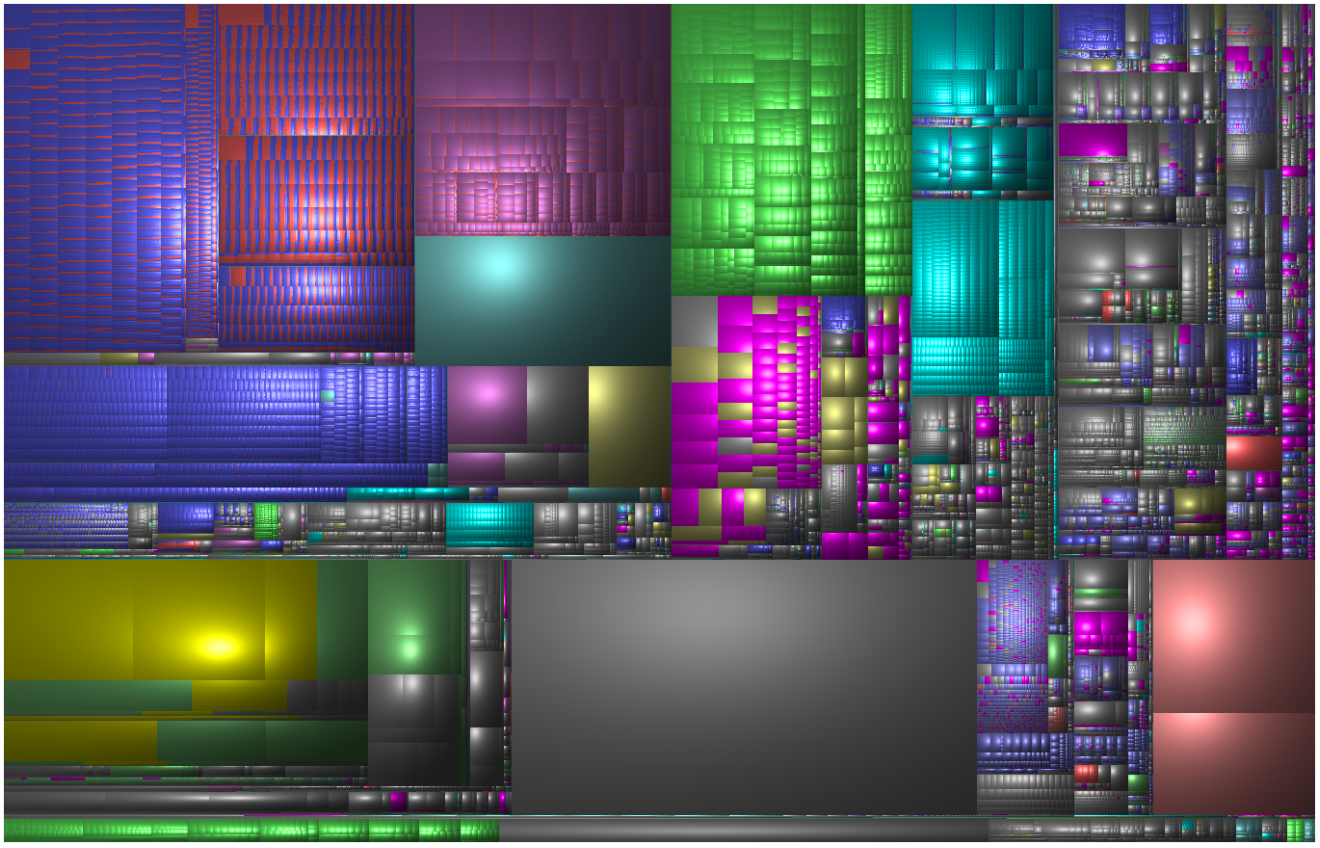


Figure 4.4: Treemap visualization of a file system.

More precisely, the tree representing the hierarchical data is laid out on a hyperbolic plane. The corresponding hyperbolic geometry has the practical properties that parallel lines diverge and that the circumference of a circle on a hyperbolic plane grows exponentially with its radius, thus providing exponentially more space for areas at outer parts of the circle. Since trees also grow exponentially in size with their depth, they can be uniformly laid out on a hyperbolic plane.

The Hyperbolic Browser typically assigns each node of the tree the same amount of angular extent on the hyperbolic plane to place its descendants in. The children of a node are then placed at equal distance to their parent node and to their siblings, along an arc spanning the designated angular extent. Laying out each node of the tree in this way, the regions assigned to each set of descendants will not overlap.

After having laid out the tree on a hyperbolic plane, it has to be projected into the two-dimensional Euclidean space in order to be of any perceivable value. There exists a number of mappings for this purpose, each of which compromises one or more aspects of the representation, like length, area, or angle. For the Hyperbolic Browser, Lamping et al. opted for the *Poincaré mapping* from the hyperbolic plane to the Euclidean unit disk, because it produced the most expedient results as it uses screen size most efficiently for the hierarchical kind of input data. Among other properties, which are detailed

in [Walter and Ritter, 2002], the Poincaré mapping is a conformal mapping that preserves angles, but distorts lines on the hyperbolic plane to arcs on the Euclidean unit disk.

Having applied the conformal mapping, the root node of the tree is represented by the center of the unit disk, while nodes at deeper hierarchy levels perspectively diminish – the more, the farther away from the center they are located. However, the user can focus on any region of the tree simply by clicking on it. In this case, a geometric translation is calculated that brings the desired point into the center of the visualization and thus magnifies the region surrounding the selected point.

Figure 4.5 illustrates the Hyperbolic Browser by depicting two screenshots generated with *music-trails* [mtr, 2008], an artist recommendation service that visualizes artist relations extracted from last.fm. This service makes use of a JavaScript implementation of the Hyperbolic Browser, cf. [hyp, 2008]. The left image shows a tree of artist relations using the band *Die Ärzte* as seed, whereas the illustration depicted on the right shows a Hyperbolic Tree whose root is the band *Black Sabbath*. In the right screenshot, the user has shifted the focus away from the root node, to the artist *Cozy Powell*. The resulting magnification of the region surrounding *Cozy Powell* and the demagnification of the regions to the lower left, which are farther away from the selection, can be clearly made out.

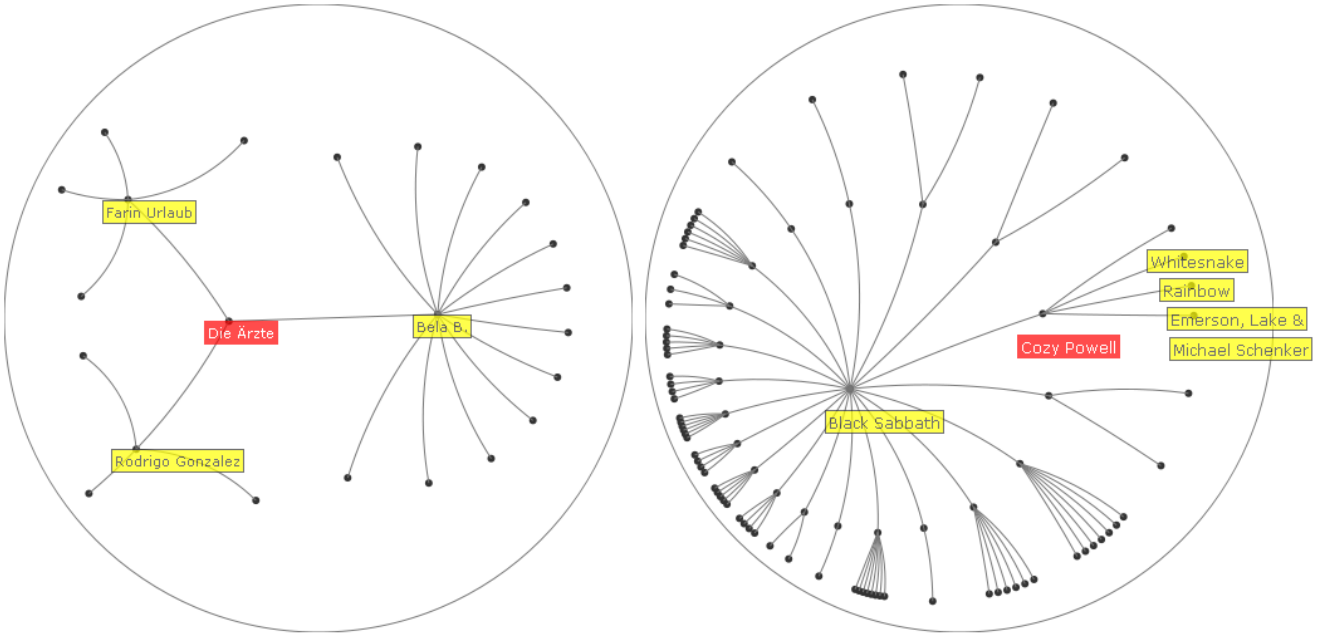


Figure 4.5: Hyperbolic Tree visualizations of similarity relations for the bands *Die Ärzte* and *Black Sabbath* as extracted from last.fm.

4.2.3 Circled Fans

While quite similar to the Hyperbolic Browser, the *Circled Fans* visualization technique developed by the author of this thesis and integrated in the CoMIRVA framework uses standard Euclidean geometry

to visualize tree-like data structures as well as networks. In addition, the Circled Fans also support illustrating a – possibly asymmetric – similarity matrix. In this case, the root element of the visualization corresponds to one entity, e.g., a music artist, represented by a row (or column) of the similarity matrix, the elements one level deeper in the hierarchy are the entities most similar to the root element, and the elements another level further down in the hierarchy are the nearest elements to the entities on the second level. When using an asymmetric similarity matrix as data source, this layout provides an easy means of differentiating between entities whose similarity relation is of a rather symmetric nature and entities with a rather unidirectional relationship.

Figure 4.6 illustrates the Circled Fans technique by depicting a visualization based on an artist similarity matrix that has been created from co-occurrences of artist names on Web pages (cf. Subsection 3.3.1). The root artist, the German hip-hop band *Die Fantastischen Vier*, is surrounded by its most similar artists according to the probability-based similarity measure. Those artists are connected to the root via straight lines of different thickness and color according to the similarity values. Using size and color as a means of information encoding, in addition to the pure textual similarity values, further helps easily discerning the most important connections, i.e., those with highest similarity values. Furthermore, the user can adjust some parameters of the visualization, like the maximum thickness of the connecting lines, the maximum number of adjacent elements (adjustable for each level separately), and the angular extent occupied by each group of child elements at the third hierarchy level. Elements at the second level are always assigned the complete 360 degrees.

A shortcoming of the Circled Fans technique is the restriction of the visualization capabilities to three hierarchy levels due to the space limitations imposed by the use of Euclidean geometry. This is to some extent alleviated by the easy browsing facilities provided. The user can create a new view on the data simply by clicking on any label to bring the selected entity in the center position and thus use it as root node of the illustration.

From Figure 4.6 it can be seen that the most similar artists to *Die Fantastischen Vier* are *Blumentopf* and *50 Cent*, followed by *Thomas D* (himself a member of *Die Fantastischen Vier*), *Busta Rhymes*, *Dr. Dre*, and *Onyx*. Moreover, the figure reveals some interesting information about the asymmetry of the similarity relations. Indeed, considering the third hierarchy level shows that only *Thomas D* and *Blumentopf* have a rather symmetric similarity relation to the root artist *Die Fantastischen Vier* as only these two hip-hop artists contain the root in their set of most similar artists. All other neighbors of *Die Fantastischen Vier* do not link back to the root artist. The most apparent explanation for this fact is that *Die Fantastischen Vier*, *Thomas D*, and *Blumentopf* form a cluster of German hip-hop artists, whereas the other artists are citizens of the United States of America and seem to have much stronger ties to their own compatriots than to their German counterparts.

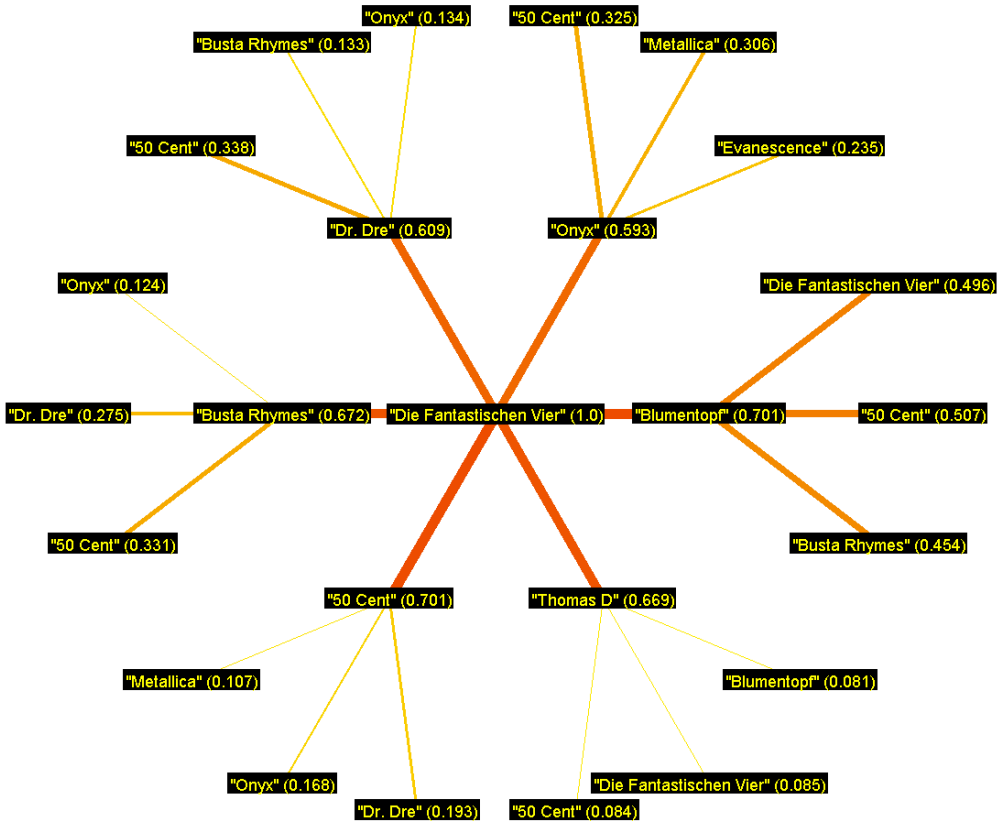


Figure 4.6: Circled Fans visualization of similarity relations for the band *Die Fantastischen Vier* derived from band name co-occurrences on Web pages.

4.2.4 Sunburst

The *Sunburst* as proposed in [Andrews and Heidegger, 1998] and [Stasko and Zhang, 2000] is a circular, space-filling visualization technique for illustrating hierarchical data. It is sometimes also referred to as *InterRing* [Yang et al., 2002]. The center of the visualization represents the highest element in the hierarchy, whereas elements on deeper levels are illustrated by arcs further away from the center. Child elements are drawn within the angular borders of their parent, but at a more distant position from the center, cf. Figure 4.7. In almost all scientific publications related to the Sunburst, its usual application scenario is browsing the hierarchical tree structure of a file system. In this scenario, directories and files are represented by arcs whose sizes are proportional to the sizes of the respective directories/files.

Unlike for the Treemap, the author is not aware of considerable extensions to the original Sunburst technique. This may be explained by the fact that the Treemap exists one decade longer than the Sunburst. To the best of the author's knowledge, only very few systems employing the Sunburst technique were published. One exception is [Keim et al., 2005], where the authors use the Sunburst technique to visualize frequent patterns for the purpose of analyzing the content of shopping carts. This application

area is quite similar to the one addressed by the Co-Occurrence Browser, cf. Subsection 4.2.6, in that the underlying co-occurrence data, from which the most important sets of terms according to a term weighting measure are extracted, can also be interpreted as frequent patterns.

Compared to the Treemap, cf. Subsection 4.2.1, the Sunburst offers the advantage of more clearly revealing the structure of the hierarchy as it displays all elements that reside on the same hierarchy level on the same torus, whereas all elements that reside on different hierarchy levels are depicted clearly visually separated on different toruses. As a result, the Sunburst frequently performs better in user studies on typical file/directory search and analysis tasks, both in time and correctness, cf. [Stasko et al., 2000]. However, the principal weakness of the Sunburst is the very small arc size that makes small elements hardly perceivable. This drawback will be addressed in the author's approach of Stacked Three-Dimensional Sunbursts, which is presented in the following.

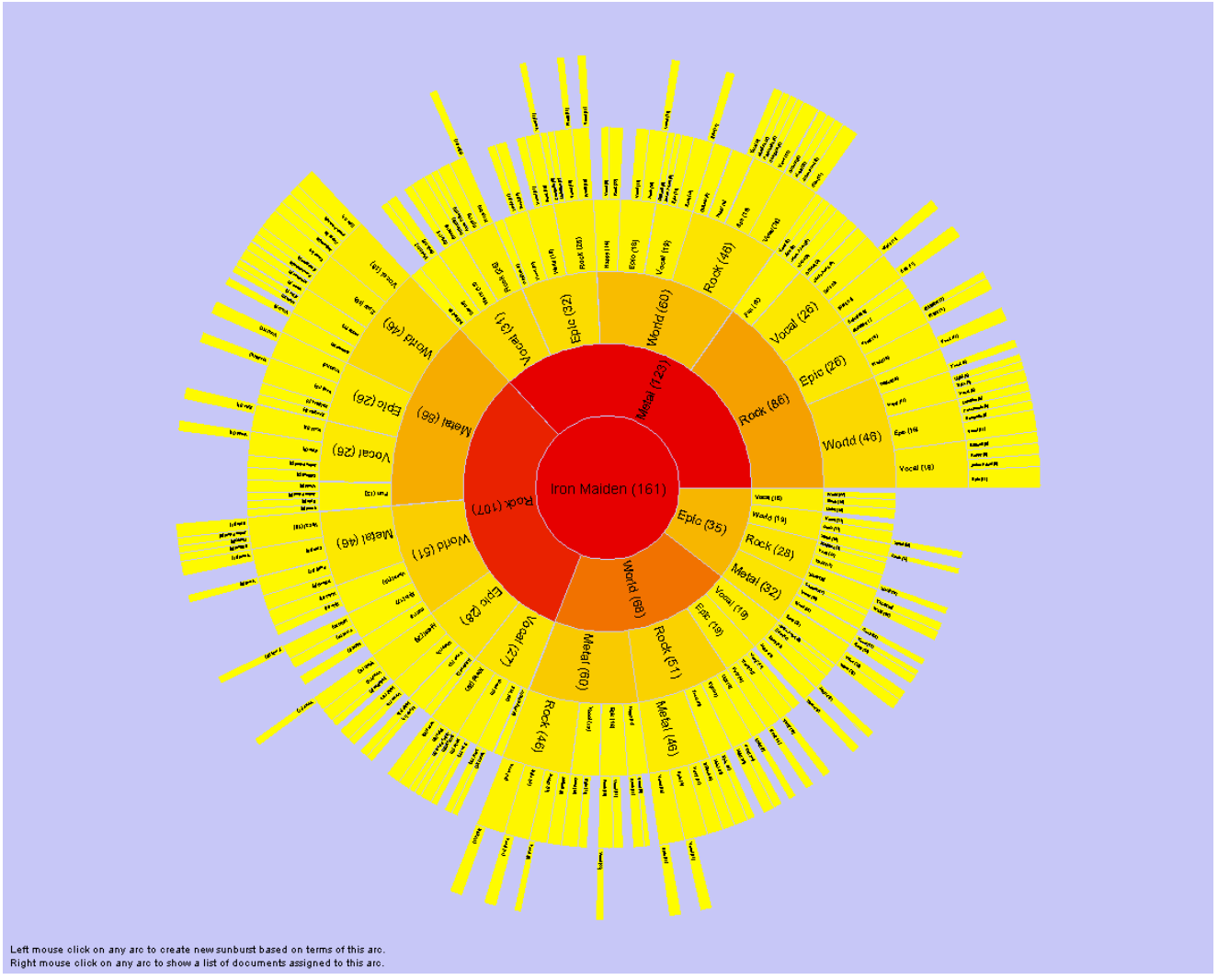


Figure 4.7: Sunburst visualization of frequently co-occurring terms on Web pages about the band *Iron Maiden*.

4.2.5 Stacked Three-Dimensional Sunbursts

In the context of this PhD thesis, a novel visualization technique for hierarchical data has been developed. The main motivation was the restriction of the original Sunburst technique. Although the Sunburst as proposed in [Andrews and Heidegger, 1998] and [Stasko and Zhang, 2000] has been proven to be an efficient space-filling visualization technique, cf. [Stasko et al., 2000], when it comes to visualizing hierarchically organized, multi-faceted data, it cannot be used since the original version allows only for encoding two different dimensions of the data. In the original application scenario described in [Stasko and Zhang, 2000], the aim is to visualize the contents of a file system. The two data dimensions which are encoded in the visualization are the sizes of files and directories (encoded as angular extent of the arcs) and the file types (encoded as arc color).

The principal improvement of the novel *Stacked Three-Dimensional Sunbursts* approach over the existing Sunburst visualization techniques is the supplementary dimensions available for visualizing additional information. Visualizing *additional dimensions of the data set* is achieved first by enhancing the visual representation of the Sunburst with a third dimension, which provides the possibility to encode continuous data in the height of each arc. Second, by organizing a set of l such enhanced three-dimensional Sunbursts in a stack, it is possible to display, for every arc, the values of l different data dimensions. Furthermore, three *constraints* for limiting the size of the Sunbursts are introduced. This is necessary as the underlying tree creation algorithm may produce trees of very large or even infinite depth and furthermore, w.l.o.g., produces trees that grow exponentially. Also the tree creation algorithm of the application described in the next section, the COB, would yield Sunbursts of extremely large sizes without constraining the generated visualization. To this end, the Sunbursts are limited with respect to the maximum number of sub nodes per node ($max\_subnodes$), the maximum depth of the hierarchy (max_depth), and the minimum angular extent of an individual arc in degrees (min_agl_ext). This last constraint was also introduced to prevent displaying arcs that are barely perceivable, a common criticism of the Sunburst technique. Using the proposed Stacked Three-Dimensional Sunbursts approach in application areas different from the one described in Subsection 4.2.6, such constraints may not be necessary.

To get a first impression of the appearance of a visualization employing the proposed approach, the reader is invited to take a look at Figure 4.8. This figure shows a Stacked Three-Dimensional Sunbursts visualization with three layers. A randomly generated data set was used, and the constraining variables were assigned the following values: $max\_subnodes = 6$, $max\_depth = 8$, $min\_agl\_ext = 4.0$. For this screenshot, no labels were generated and drawn to avoid visual clutter. To emphasize the data dimension represented by the angular extent of the arcs, a colormap was applied to the values of this data dimension and used to color each arc accordingly.

Visualizing Various Data Dimensions

Each arc of one fixed Sunburst in the stack is capable of encoding three different data dimensions. The first one is represented by the arc's angular extent, which obviously has to be the same for a particular arc over all layers of the stack (otherwise the stacking would not make much sense). The second one is encoded in the height of each arc, which may vary for a fixed arc according to the layer. In addition, color may be used to emphasize one of these two dimensions, or to encode a third data dimension. However, the author does not recommend visualizing a third data dimension by means of different coloring since this would most probably lead to heavy confusions of the users. Instead, color should be used as a means of emphasizing, as illustrated in Figures 4.8 and 4.9. Figure 4.8 illustrates the use of color to emphasize the data dimension encoded in the angular extent, whereas in Figure 4.9 different coloring serves to emphasize the dimension encoded in the height. In total, considering a Sunburst stack of l layers and assuming that color is not used to visualize an additional data dimension, it is possible to encode the values of $l+1$ data dimensions for each arc (one in the arc's angular extent and l in its height in each of the l layers).

Labeling

For each arc A, a label can be displayed, which is drawn above A and colored with the same color as used for A. The label has the same angular orientation as the corresponding arc in order to facilitate assigning it to its arc. Unfortunately, this comes at the cost of seeing labels of arcs with an angular extent of more than 180 degrees from behind. Thus, the author suggests rotating such labels by 180 degrees. As for the label size, it is automatically adjusted to the maximum possible size that fits within the borders of the corresponding arc.

User Interaction

The author proposes the following ways of navigating in a Stacked Three-Dimensional Sunbursts visualization, which are also provided in a prototypical Java implementation. Obviously, *rotating* the Sunburst stack around the Y-axis, i.e., the vertical axis going through the root nodes of all Sunbursts in the stack, should be offered. This is provided by moving the mouse in the horizontal direction while pressing an arbitrary mouse button. *Zooming* in/out (within predefined boundaries) is accomplished by holding the left mouse button pressed and moving the mouse upwards/downwards. Rotation around the X-axis, i.e., *changing the inclination of the stack* is provided, but limited to angles between a front view and a bird's eye view. This function is supported by moving the mouse upwards/downwards while holding the right mouse button pressed.

As for selecting a particular arc, e.g., to display additional information about the corresponding data item, the prototypical implementation allows to use the cursor keys to navigate in the hierarchy. Using the keys *arrow down* and *arrow up*, the hierarchy is browsed in a vertical manner. More precisely, with the *arrow down* key, the first child arc of the currently selected arc is chosen, while the *arrow up* key selects the parent of the currently selected arc. Using the keys *arrow left* and *arrow right*, the user can navigate within the elements on the same hierarchy level which are grouped by the selected parent arc. The currently selected arc is highlighted by means of drawing a white border around it and coloring its label in white. Depending on the application scenario for which the visualization is used, the author suggests two methods for highlighting arcs. The first one highlights only the most recently selected arc and should be used when it is important to concentrate the users attention on this arc. Employing the second method, the selected arc at each hierarchy level is highlighted, which facilitates tracing the selection back to the root arc. This second highlighting method is used in the COB application, which is presented in the following.

4.2.6 Co-Occurrence Browser

The technique proposed in the previous subsection can be applied to a wide area of application, provided that the data to be visualized is hierarchically structured, and every element in the hierarchy is assigned a set of attributes. Possible application areas hence include the one proposed in [Stasko and Zhang, 2000], i.e., illustrating the hierarchical tree structure of a file system. In this case, the attributes assigned to each element (file/directory) may be *time elapsed since the last change of the file* or *number of file accesses*. Furthermore, the Stacked Three-Dimensional Sunbursts may be employed to visualize the product hierarchy of (Web) shops offering a large range of products or that of online auction systems like *eBay* [ebay, 2007]. In this case, one Sunburst layer for each of the attributes *price of the product* or *current bid*, *time remaining until the end of the auction*, *different feedback levels (positive, neutral, negative)* or *distance from the item location to the user's own domicile* may be included in the visualization.

To stay within the scope of this thesis, however, the author elaborated a novel application for the scenario of exploring a collection of Web pages about music artists. This application is called the *Co-Occurrence Browser* (COB) and employs the Stacked Three-Dimensional Sunbursts technique. COB's purpose is threefold. First, the COB facilitates getting an *overview* of the Web pages related to a specific music artist, by structuring them according to co-occurring terms. Second, since the terms that most often occur on Web pages related to a music artist X constitute an individual profile of X, the COB is also suited to reveal various artist-related *meta-information* in the form of descriptive terms, cf. Subsection 2.2.1, e.g., musical style, related artists, or instrumentation. Third, by visualizing

the amount of *multimedia content* provided at the indexed Web pages, the user is offered a means of exploring the audio, image, and video content of the respective set of Web pages.
The COB has been implemented in Java using the *processing* environment [pro, 2007] and builds upon the author's *CoMIRVA* framework for music and multimedia processing, information retrieval, and visualization, cf. [Schedl et al., 2005d] or [Schedl et al., 2007a].

Creation of the Sunbursts

Using the inverted index of the Web pages of an artist X, as given by the procedure described in Subsection 3.2, it is easy to extract subsets $S_{X,\{t_1,\ldots,t_d\}}$ of the Web page collection of X which have in common the occurrence of all terms $t_1, \ldots, t_d$.
In the application scenario of the COB the size of the Sunburst is in principle only restricted by the number of possible combinations of all terms used for indexing. Thus, a set of stop criteria for complexity limitation, as already indicated in Subsection 4.2.5, is needed. These constraints are detailed below.
Starting with the entire set of Web pages $S_{X,\{\}}$ retrieved for an artist X, a maximum number l of terms with highest weights according to the applied term weighting measure are used to create l subsets $S_{X,\{t_1\}}, \ldots, S_{X,\{t_l\}}$ of the collection.[1] These subsets are visualized as arcs $A_{X,\{t_1\}}, \ldots, A_{X,\{t_l\}}$ around a centered cylinder which represents the root arc $A_{X,\{\}}$, and thus the entire set of Web pages retrieved for artist X. The angular extent of each arc is proportional to the term weight of the associated term t_i. To avoid very small arcs that are hardly perceivable, arcs whose angular extent is smaller than a fixed threshold that equals the parameter min_agl_ext, introduced in Section 4.2.5, are omitted. Furthermore, each arc is depicted in the color obtained by applying the colormap "Sun" from the CoMIRVA framework to the relative weight of the arc's corresponding term t_i (relative to the maximum weight).
The term selection with respect to term weights and the corresponding visualization step are recursively performed for all arcs, with a maximum max_depth for the recursion depth. This eventually yields a complete Sunburst visualization, where each arc at a specific recursion depth d represents a set of Web pages $S_{X,\{t_1,\ldots,t_d\}}$ in which all terms $t_1, \ldots, t_d$ co-occur.
Internally, the Sunburst is stored as a tree, where each arc is represented by a node. A node $A_{X,\{t_1,\ldots,t_d\}}$ at depth d in the tree thus represents the set of Web pages that contain the term t_d and all other terms $t_1, \ldots, t_{d-1}$ associated with the nodes on the shortest path from $A_{X,\{t_1,\ldots,t_d\}}$ to the root node.
As for the different layers in the Sunburst stack, each layer illustrates the amount of a specific category of multimedia files that are linked or embedded in the Web pages under consideration. Information

[1] The COB uses document frequencies for term weighting. This choice results from a user study which is elaborated in Subsection 5.3.3.

on these files is extracted as described in Subsection 3.2. The three categories are audio, image, and video. To this end, the COB encodes the relative number of the audio, image, or video files in the height of the arcs (relative to the total number represented by the root node of the respective layer). For example, denoting the audio layer as L_A, the image layer as L_I, and the video layer as L_V and focusing on a fixed arc A, the height of A in L_I shows the relative number of image files contained in the Web pages that are represented by arc A, the height of A in L_V illustrates the relative number of video files, and the height of A in L_A the relative number of audio files.

User Interface

Figure 4.10 depicts a screenshot of COB's user interface for 198 Web pages retrieved for the band *Iron Maiden*. The clustering according to co-occurrences of terms with highest weights, given by their document frequencies, was performed as described in the last subsection. Each arc $A_{X,\{t_1,...,t_d\}}$ is labeled with the term t_d that subdivides the Web pages represented by the arc's parent node $A_{X,\{t_1,...,t_{d-1}\}}$ into those containing t_d and those not containing t_d. Additionally, the document frequency of the term t_d is added in parentheses to the label of each arc $A_{X,\{t_1,...,t_d\}}$. In the Stacked Three-Dimensional Sunbursts visualization shown in Figure 4.10, the topmost layer illustrates the amount of video files indicated on the Web pages, the middle layer the amount of image files, and the lower layer the amount of audio files. In this screenshot, the arc representing the Web pages on which all of the terms "metal", "Iron Maiden", and "guitar" co-occur is selected. For 74 out of the complete set of 198 Web pages, this is the case.

Extended Functionalities for User Interaction

In addition to the user interaction functionalities proposed in Subsection 4.2.5 for the Stacked Three-Dimensional Sunbursts, the following interaction possibilities are provided to fully exploit the features offered by the COB.

- Pressing the key *Return/Enter* creates a new visualization only including the Web pages of the selected arc.
- The *Backspace* key restores the original visualization, whose root node represents all Web pages retrieved for the artist under consideration.
- The key *D* lists the URLs of the Web pages that are represented by the selected arc.
- Pressing the keys *A*, *I*, or *V* shows a list of audio, image, or video files, respectively, which are found on the Web pages of the currently selected arc.

- Using the keys *W* and *S*, it is possible to browse in the currently displayed list of URLs, audio files, image files, or video files.
- Pressing *C* toggles the data dimension which is encoded in the color of the arcs, between the number of Web pages and the amount of multimedia data.

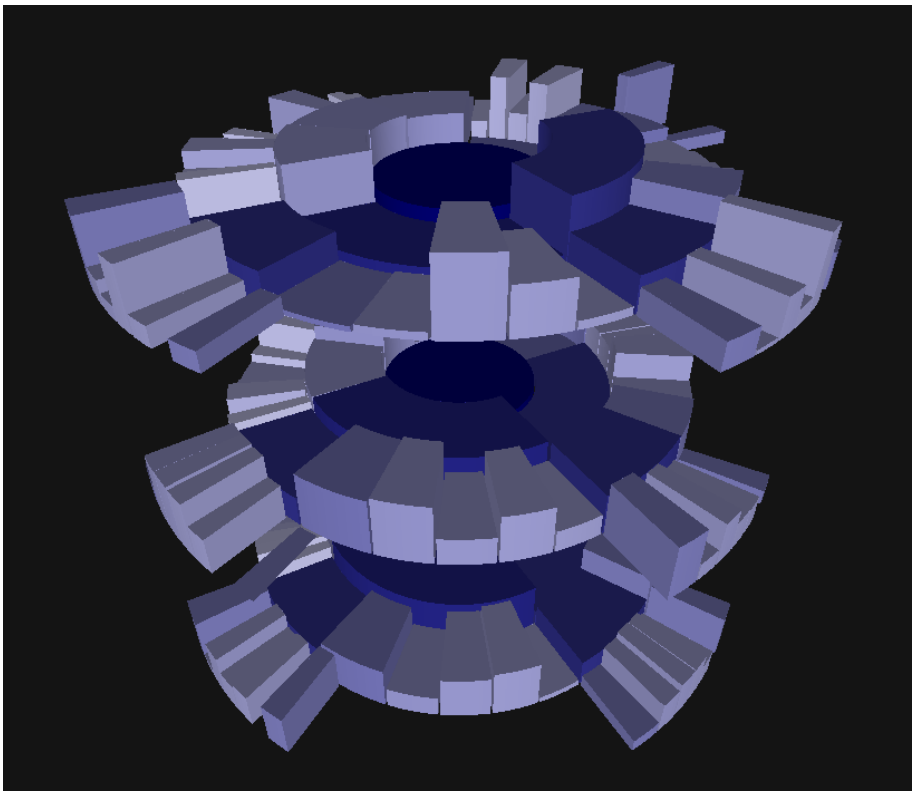


Figure 4.8: Stacked Three-Dimensional Sunbursts visualization using three layers. Color is used to emphasize the data dimension encoded in the arcs' angular extent.

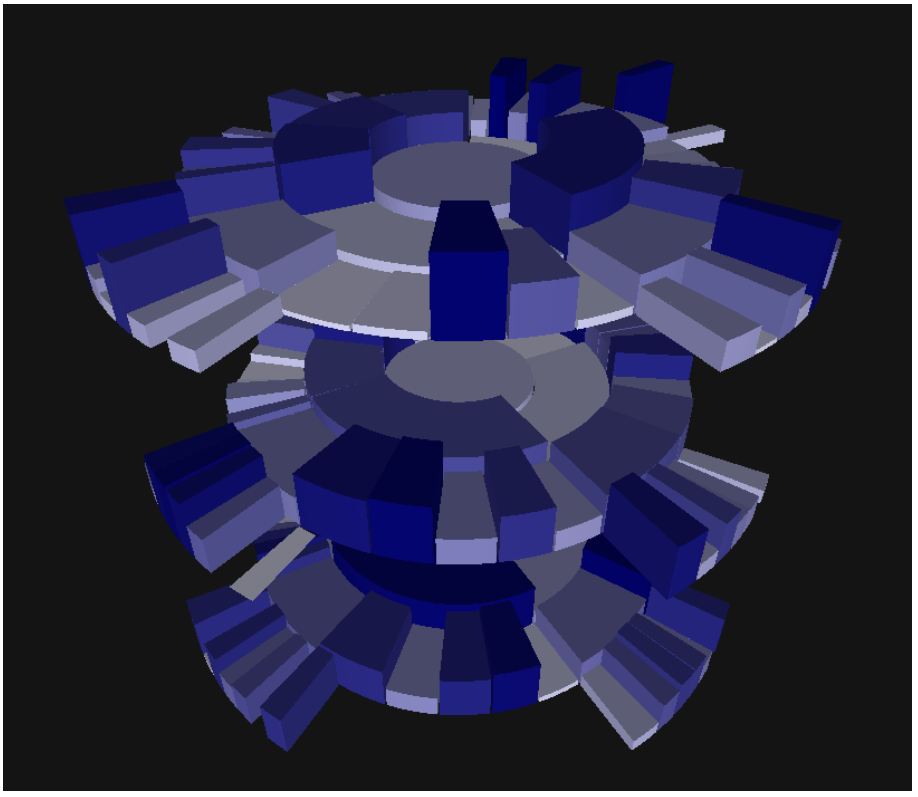


Figure 4.9: Stacked Three-Dimensional Sunbursts visualization using three layers. Color is used to emphasize the data dimensions encoded in the arcs' heights.

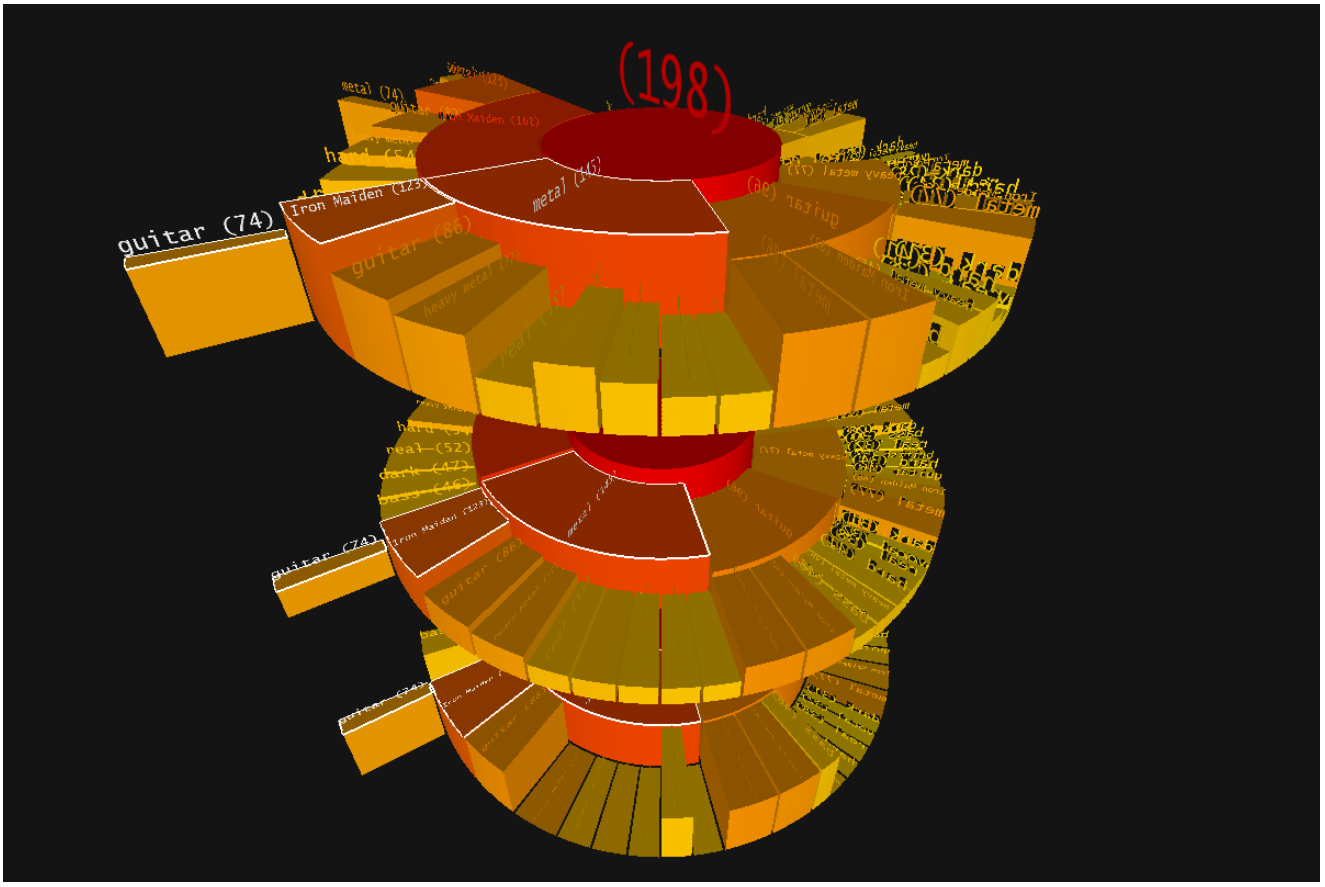


Figure 4.10: COB used to illustrate 198 Web pages of the band *Iron Maiden*, which are clustered according to co-occurring terms.

Chapter 5

Experiments and Evaluation

In this chapter, first, the collections that defined the ground truth in the various experiments performed are briefly presented. Second, the different query schemes used in the process of Web page retrieval are outlined. Third, the conducted experiments are elaborated, grouped into different subsections according to the information retrieval or visualization task that is addressed by the approach under evaluation. The general structure of the subsections dealing with concrete experiments is given by an introductory description of the experimental setup, followed by a description of the experiment itself, and finally, concluded by the presentation and discussion of the results.

5.1 Test Collections

In the following, an overview of the test collections used for evaluation is given. Due to the different kinds of extracted information and to a missing large collection that comprises all these information categories, different test collections, which are tailored to the information category under evaluation, are required. Furthermore, it is desirable to evaluate some approaches on collections of different difficulty, e.g., different number and granularity of genres used in genre classification experiments or varying popularity of artists in the evaluation of prototypicality estimation approaches.
The variety of test collections used in this PhD thesis necessitates the following standardized denotation scheme: All collections will be named starting with a *"C"*, followed by the number of artists (or bands) and an *"a"* or the number of albums and a *"b"*, optionally followed by the number of genres and a *"g"*, optionally followed by the number of band members and an *"m"*.

The collection *C224a14g* is mainly used for evaluating artist similarity and artist prototypicality estimation approaches. It comprises 14 quite general genres, each consisting of 16 popular artists that are easily assigned to their respective genre without much ambiguity. To assess the descriptiveness of terms for particular artists, a subset *C112a14g* of *C224a14g* was extracted. The composition of both collections can be found in Tables A-1 to A-3 in the appendix.
In contrast, the collections *C103a22g* and *C103a13g*, whose main purpose is also artist similarity evaluation, contain only 103 artists. While *C103a22g* categorizes these artists into 22 genres of highly

varying specificity (including general ones like "Blues" and "Jazz" as well as very specific ones like "Melodic Metal" and "German Hip Hop"), the collection entitled *C103a13g* shows a more general genre structure as it clusters together very narrow genres such as the various subgenres of "Metal". The composition of the collections can be found in Tables A-4 to A-7.

The collection *C1995a9g* has been extracted from AMG to evaluate the proposed artist prototypicality estimation approaches. It contains 1,995 artists categorized into 9 very general genres. The popularity of the included artists varies from very popular to barely known. Due to its size, *C1995a9g* cannot be listed here exhaustively, but the distribution of genres and AMG tiers is depicted in Table A-8. *C1995a9g* is a subset of the artists available in AGMIS, cf. Section 6.1, for which information on tiers are provided by AMG.

The test sets for evaluating band members and instrumentation detection are made up of the collections *C51a240m* and *C51a499m*, whose artist names can be found in Table A-9. Both were compiled from the author's private music collection and mainly comprise rock and metal bands of very different popularity. *C51a240m* includes the 240 current members of the 51 bands under consideration. *C51a499m* is an extended version of *C51a240m*. It includes not only currently active band members, but also all former members.

For evaluating the proposed album cover retrieval techniques, the two collections *C255b* and *C3311b* are used. *C255b* is a subset of the author's private compact disc collection and contains 255 album names from 118 distinct artists. It features mainly European and, to a smaller extent, American artists. A list of all artist and album names in *C255b* can be found in Tables A-10 to A-14. In contrast, *C3311b* is a larger, commercial collection comprising 3,311 album names from 1,593 distinct artists from all around the world, even from China. Due to space limitations, the exact composition of *C3311b* cannot be given here.

5.2 Query Schemes

The main challenge of the employed Web page retrieval approach, i.e., querying a search engine with constrained queries, cf. Subsection 3.1, is to ensure obtaining **relevant** Web pages, both with respect to the artist under consideration and to the task that is to be solved. This is addressed by using different query schemes, which are tailored to direct the search towards *music-related* pages, in general, and to pages that contain the *specific information* requested by the respective information retrieval problem, in particular. In Table 5.1, the query schemes used in the experiments are presented, together with an abbreviation for each. Here, the schemes are depicted using the Google syntax for query formulation.

Abbreviation	Query Scheme
M	"*artist/band name*"+music
MR	"*artist/band name*"+music+review
MGS	"*artist/band name*"+music+genre+style
AIT	allintitle:"*artist/band name*"
MM	"*artist/band name*"+music+members
LUM	"*artist/band name*"+lineup+music
C	"*artist/band name*"+"*album name*"+cover
CA	"*artist/band name*"+"*album name*"+cover+album

Table 5.1: Query schemes used in the experiments conducted.

5.3 Experiments

In the following, the experiments conducted to evaluate the approaches proposed in Sections 3 and 4 are described. If not stated otherwise, the Web search engine provided by Google was queried for the approaches involving such a step.

5.3.1 Artist Similarity

Evaluating the use of Web mining techniques for the task of deriving similarities between music artists has been performed on the collections *C224a14g* and *C103a22g* using the methods *co-occurrence analysis based on page counts*, *co-occurrence analysis based on retrieved page content*, and *deriving similarities from artists' term weighting vectors (TF·IDF)*, which are respectively described in Subsections 3.3.1, 3.3.1, and 3.3.3.

Regardless of whether co-occurrence analysis is performed using only information on page counts or searching for artist names in Web page content, one could assume that this simple approach, which is, in fact, equivalent to computing relative document frequencies of artist names on Web pages, is considerably outperformed by the standard text retrieval approach of calculating similarities on TF·IDF vectors. Therefore, the main objective of the evaluation was to compare the performance of these two approaches.

Since direct quantization of similarity between music artists is infeasible due to the various dimensions of music similarity, which are – even worse – often perceived differently by different subjects[1], the concept of genre is typically used to cluster similar artists according to their musical style. To this end, first, experiments were run on *C224a14g*, and performance was measured using the *ratio between intra- and intergenre similarities* as well as *k-NN classification with cross-validation*. The aim was to estimate how well the given genres are distinguished by the similarity measures under evaluation. To evaluate

[1] One may think of the mood attributed to a piece of music, for example. Also the perception of tempo is likely to differ between a jazz lover and a listener of speed metal.

how well the proposed approaches perform on a collection structured according to a more specific and finer grained genre taxonomy, k-NN classification experiments were then conducted on the collection *C103g22*. The higher specificity of the genres in *C103g22* results in less popular artists, which is also reflected by the total number of Web pages available, on average, for each artist (150,980 pages for *C224a14g* vs. 66,955 pages for *C103g22* when querying Google with the MR scheme).

Intragenre/Intergenre Similarities

Employing this method, for each genre, the fraction between the *average intragenre similarity* and the *average intergenre similarity* is calculated. The higher this ratio, the better the discrimination of the respective genre. The average intragenre similarity for a genre g is the expected similarity between two randomly chosen artists a and b from genre g. The average intergenre similarity for a genre g, in contrast, is the expected similarity between two randomly chosen artists a (from genre g) and b (from any other genre).

Formally, let A be the set of all artists in the collection, and let A_g be the set of artists assigned to genre g. The average intragenre and intergenre similarities are then given by Formulas 5.1 and 5.2, respectively, where $|A_g|$ is the cardinality of A_g and $A \setminus A_g$ is the set A without the elements contained in the set A_g. The ratio $intra_g/inter_g$ should obviously be at least greater than 1.0 if the similarity measure is to be of any use.

$$intra_g = \frac{\sum_{a \in A_g} \sum_{b \in A_g}^{b \neq a} sim(a,b)}{|A_g|^2 - |A_g|} \tag{5.1}$$

$$inter_g = \frac{\sum_{a \in A_g} \sum_{b \in A \setminus A_g} sim(a,b)}{|A \setminus A_g| \cdot |A_g|} \tag{5.2}$$

Results and Discussion

Table 5.2 shows the results of evaluating the co-occurrence approach based on page counts using intragenre/intergenre ratios. It can be seen that the AIT scheme yields the best results as the average intergenre similarities are very low. Hence, nearly no artists from different genres occur together in the title of the same Web page. Especially for the genres "Jazz" and "Classical", the results are remarkable. However, for "Alternative Rock / Indie" and "Electronica", the ratios are quite low. This can be explained by the low average intragenre similarities for these genres. Thus, artists belonging to these genres are seldom mentioned together in titles. Analyzing the page count matrices reveals that the AIT scheme yields good results if there exists Web pages containing artists from the same genre in their title. However, 87.88% of the elements in the page counts matrix created using the AIT scheme

have a value of zero.[2] Since the artists in collection *C224a14g* are very popular, one can assume that this percentage is even higher for collections comprising less known artists. Therefore, the AIT scheme does not seem suitable for real world collections. Among the other schemes, MR yields overall better results than M. Moreover, Table 5.2 shows that, aside from "Classical", "Blues" is distinguished quite well. Also remarkable is the very bad result for "Folk" when using the MR scheme. This may be explained by intersections with other genres, in particular, with "Country".

To compare the approach to co-occurrence analysis proposed in Subsection 3.3.1 with the one proposed in [Zadel and Fujinaga, 2004], intragenre/intergenre similarities were also calculated employing the method by Zadel and Fujinaga. The results are depicted in Table 5.3. They are, overall, slightly worse than those shown in Table 5.2 for the same data set *C224a14g*. The reason for this is probably the different normalization method used in [Zadel and Fujinaga, 2004], which discards any information on the asymmetry of the page counts matrix.

Table 5.4 shows the evaluation results for the co-occurrence approach based on content analysis, as described in Subsection 3.3.1. The results are, in general, better than those obtained with the co-occurrence analysis based on page counts. This may be due to the fact that the page counts offered by Google are not deterministically calculated values, but only estimates of the real page counts. Thus, it seems that analyzing the Web pages' content, even if only performed for the 100 top-ranked pages, gives more accurate results than relying on Google's estimates.

In Table 5.5, the evaluation results for the TF·IDF-based approach to similarity computation, which was elaborated in Subsection 3.3.3, are depicted. Taking a closer look at these results shows that TF·IDF performs better for eight genres, whereas co-occurrence analysis based on page counts performs better for six genres. However, the mean of the ratios is higher for the co-occurrence approach, because of the high ratio for the genre "Classical". A possible explanation is that Web pages about classical artists often also contain words which are used on pages of other genres' artists. In contrast, classical artist names seem to be mentioned mostly together with other classical artist names, which is reflected by the very high ratios of the co-occurrence approach for this genre. Interestingly, taking into consideration all terms that occur on a set of Web pages, as in the case of the TF·IDF approach, does not increase the intragenre/intergenre performance when compared with the co-occurrence approach based on content analysis, which does only investigate artist names. This finding is consistent with the assumption that a specialized dictionary of music terms, cf. Subsection 3.2, is likely to increase the performance of term weighting approaches, while at the same time reduces computational complexity.

[2] In contrast, this number is only 0.44% for the MR scheme and 0.0045% for the M scheme.

Query Scheme	MR			M			AIT		
Genre	$intra_g$	$inter_g$	$ratio$	$intra_g$	$inter_g$	$ratio$	$intra_g$	$inter_g$	$ratio$
Country	0.088	0.032	2.723	0.104	0.039	2.644	2.170E-3	3.092E-5	70.163
Folk	0.052	0.098	0.529	0.054	0.039	1.374	5.877E-4	2.453E-5	23.963
Jazz	0.094	0.038	2.460	0.132	0.039	3.399	5.052E-3	2.377E-5	212.534
Blues	0.132	0.026	5.085	0.106	0.024	4.483	2.058E-3	2.377E-5	58.496
RnB / Soul	0.068	0.032	2.148	0.078	0.044	1.780	9.400E-4	3.785E-5	24.839
Heavy Metal / Hard Rock	0.208	0.126	1.649	0.267	0.083	3.206	8.808E-4	5.708E-5	15.432
Alternative Rock / Indie	0.091	0.072	1.261	0.191	0.079	2.426	3.733E-4	8.822E-5	4.232
Punk	0.139	0.098	1.419	0.192	0.067	2.860	1.109E-3	2.300E-5	48.210
Rap / Hip-Hop	0.110	0.066	1.654	0.153	0.055	2.798	1.855E-3	7.092E-5	26.159
Electronica	0.074	0.042	1.774	0.134	0.047	2.872	4.494E-4	5.014E-5	8.962
Reggae	0.135	0.048	2.807	0.072	0.036	2.013	9.807E-4	2.847E-5	34.455
Rock 'n' Roll	0.075	0.041	1.817	0.086	0.045	1.899	1.556E-3	6.248E-5	24.907
Pop	0.134	0.066	2.040	0.178	0.072	2.470	1.501E-3	8.023E-5	18.819
Classical	0.312	0.010	31.733	0.201	0.011	18.177	1.154E-2	4.504E-6	2,561.504
Mean			**4.221**			**3.743**			**223.762**

Table 5.2: Evaluation results of intragenre/intergenre similarities using co-occurrence analysis based on page counts.

Query Scheme	MR			M			AIT		
Genre	$intra_g$	$inter_g$	$ratio$	$intra_g$	$inter_g$	$ratio$	$intra_g$	$inter_g$	$ratio$
Country	0.136	0.050	2.725	0.150	0.058	2.591	2.988E-3	5.401E-5	55.328
Folk	0.080	0.159	0.502	0.082	0.058	1.340	1.115E-3	4.294E-5	25.962
Jazz	0.129	0.059	2.273	0.180	0.056	3.235	6.585E-3	3.842E-5	171.398
Blues	0.178	0.040	4.448	0.154	0.036	4.222	3.125E-3	5.572E-5	56.080
RnB / Soul	0.097	0.050	1.950	0.107	0.065	1.655	1.180E-3	5.719E-5	20.627
Heavy Metal / Hard Rock	0.295	0.185	1.592	0.379	0.122	3.112	1.517E-3	1.095E-4	13.857
Alternative Rock / Indie	0.141	0.116	1.209	0.286	0.118	2.430	7.118E-4	1.622E-4	4.389
Punk	0.201	0.140	1.429	0.272	0.097	2.796	1.591E-3	3.888E-5	40.909
Rap / Hip-Hop	0.164	0.097	1.683	0.223	0.080	2.774	3.256E-3	1.169E-4	27.850
Electronica	0.111	0.062	1.798	0.187	0.068	2.758	6.581E-4	9.009E-5	7.305
Reggae	0.216	0.086	2.513	0.111	0.058	1.934	1.622E-3	4.808E-5	33.745
Rock 'n' Roll	0.117	0.065	1.793	0.131	0.069	1.905	2.199E-3	9.718E-5	22.630
Pop	0.210	0.107	1.952	0.257	0.112	2.302	2.316E-3	1.387E-4	16.698
Classical	0.423	0.016	26.438	0.270	0.016	16.592	1.548E-2	7.556E-6	2,048.574
Mean			**3.736**			**3.551**			**181.811**

Table 5.3: Evaluation results of intragenre/intergenre similarities using co-occurrence analysis as proposed in [Zadel and Fujinaga, 2004].

Query Scheme	MR		
Genre	$intra_g$	$inter_g$	$ratio$
Country	0.066	0.015	4.511
Folk	0.078	0.017	4.552
Jazz	0.083	0.007	12.227
Blues	0.074	0.009	8.527
RnB / Soul	0.064	0.011	5.889
Heavy Metal / Hard Rock	0.081	0.014	5.872
Alternative Rock / Indie	0.062	0.021	2.907
Punk	0.123	0.015	8.225
Rap / Hip-Hop	0.089	0.015	5.912
Electronica	0.057	0.011	5.137
Reggae	0.060	0.008	7.866
Rock 'n' Roll	0.071	0.012	5.773
Pop	0.104	0.020	5.269
Classical	0.122	0.002	54.634
Mean			**9.807**

Table 5.4: Evaluation results of intragenre/intergenre similarities using co-occurrence analysis based on retrieved page content.

Query Scheme	MR		
Genre	$intra_g$	$inter_g$	$ratio$
Country	0.118	0.049	2.425
Folk	0.064	0.043	1.480
Jazz	0.131	0.048	2.722
Blues	0.134	0.047	2.875
RnB / Soul	0.109	0.060	1.812
Heavy Metal / Hard Rock	0.080	0.049	1.618
Alternative Rock / Indie	0.075	0.049	1.521
Punk	0.098	0.053	1.848
Rap / Hip-Hop	0.129	0.050	2.545
Electronica	0.077	0.039	1.985
Reggae	0.135	0.045	3.025
Rock 'n' Roll	0.105	0.050	2.099
Pop	0.081	0.052	1.577
Classical	0.230	0.025	9.164
Mean			**2.621**

Table 5.5: Evaluation results of intragenre/intergenre similarities using TF·IDF vectors.

k-Nearest Neighbors Classification

The second set of evaluation experiments was conducted to show how well the similarity measures perform for classifying artists into genres. For this purpose, the widely used technique of *k-Nearest Neighbors classification* (k-NN) was chosen. This technique simply uses those k data items for prediction that have minimal distance to the item that is to be classified. The most frequent class among these k data items is then predicted. If this class is ambiguous, the class of the nearest item among the eligible classes is predicted. As for the partitioning of the complete data set into training set and test set, a variant of *k-fold cross-validation* is used. More precisely, a certain number x of artists from each genre is selected for training, the remaining are used for testing. For reasons of simplicity, the corresponding experiment is abbreviated as t_x. In a t_{15} evaluation experiment, for example, 15 artists from each genre are used for training and one remains for testing. All experiments were run 1,000 times to minimize the influence of statistical outliers on the overall results. *Accuracy*, in the following used to measure performance, is defined as the percentage of correctly classified data items over all classified data items in the test set. In some cases, however, no prediction is possible due to a lack of available data. Especially, when using the AIT scheme, this happens quite often.
In a first experiment evaluating co-occurrence analysis on *C224a14g* with setting t_8, k-NN with $k = 9$ performed best. It is not surprising that values around 8 perform best in a t_8 experiment, because in this case, the number of data items in the training set that are used for prediction equals the number of data items chosen from each class to represent the class in the training set. The t_8 setting gives accuracies of about 69% for the M scheme, about 59% for the MR scheme and about 74% for the AIT scheme. Using setting t_{15}, these results can be improved to about 75% for M (9-NN) and to 80% for AIT (6-NN). For the MR scheme, no remarkable improvement could be achieved.

Confidence Filter In some cases, particularly when using the AIT scheme, no artist information is available, i.e., for some artists, no Web pages mentioning any other artist in the collection can be found. In this case, the artist's similarity to all other artists in the training set is zero, and a random genre would be predicted for this artist. Due to the sparseness of its similarity matrix, this problem mainly concerns the AIT scheme. To overcome the problem and benefit from the good performance of the AIT scheme, but also address the sparseness of the respective similarity matrix, it seems reasonable to combine the similarity measures yielded by the different query schemes. The basic idea is to use the similarity measure based on the AIT scheme if the confidence in its results is high enough. If not, the M- or MR-based measure is used to classify an unknown data item. To incorporate such a *confidence filter* in the classification process, the number of elements with a similarity of zero in the set of the nearest neighbors is counted for the AIT measure. If this number exceeds a given threshold z, the

respective data item is not classified with the AIT measure, but the M or MR measure is used instead. Employing this simple method, only artists that co-occur at least with some others in the title of some Web pages are classified with AIT. On the other hand, if not enough information for a certain artist can be derived from the AIT query results, MR or M is used instead. Such compound similarity measures usually give enough information for prediction. Indeed, the percentage of artists that can be classified with the compound measure equals 100% for the collection *C224a14g*. This can also be seen in Figure 5.1, which shows that the accuracies for MR and M are largely independent of the threshold z for the confidence filter.

k-NN classification experiments were also conducted to compare the co-occurrence approach with the TF·IDF approach. For these experiments, the threshold z was set to zero, i.e., predictions were made based on the AIT measure only when similarity information could be calculated at least for the k nearest neighbors.

Results and Discussion

It was already reported that classification accuracies of up to 80% for single, uncombined measures could be achieved on the collection *C224a14g*. To analyze to what extent the performance can be improved when using compound measures, t_{15} validations using either a single measure or combinations of AIT with MR and M were performed. The results are shown in Figure 5.1. Along the abscissa, the influence of different thresholds z for the confidence filter can be seen. Although yielding the highest accuracy values, it is important to note that the single AIT measure does not always make a prediction. In fact, the high values for the AIT accuracies come along with up to 18% of unclassified artists. Nevertheless, accuracies of up to 89.5% were achieved for $z = 2$ using the AIT scheme. In this case, 14% of the artists could not be classified. Taking a closer look at the results for the query schemes MR and M shows that they reach accuracies of about 54% and 75% respectively and that these results are largely independent of the value of z. Hence, the query schemes M and MR always provide enough information to make a prediction. Combining the measures by taking AIT as primary one and, if no classification is possible, MR or M as fallback thus combines the advantages of high accuracy and high percentage of classified artists. Indeed, using the combination AIT+M gives accuracies of 85% for 100% classified artists. Since the accuracies for M are considerably higher than for MR, the combination of AIT with M yields better results than with MR.

As already mentioned, the single AIT measure using the confidence filter does not always make a prediction. To analyze the trade-off between accuracy and percentage of classified artists, in Figure 5.2, these two dimensions are plotted against each other. This figure shows that, in general, an increase

	3-NN			7-NN		
	t_2	t_4	t_8	t_2	t_4	t_8
MR	56	68	74	43	68	77
MGS	54	66	73	39	67	75

Table 5.6: Accuracies, in percent, for different training set sizes using the TF·IDF approach on collection *C224a14g*.

in accuracy goes along with a decrease in classified artists. The highest accuracies obtained for the different settings are 89% for t_{15} (86% classified artists), 84% for t_8 (59% classified artists), 64% for t_4 (34% classified artsits), and 35% for t_2 (10% classified artists). These maximum accuracy values are usually achieved with a threshold z of 1 or 2 for the confidence filter.

To investigate how many artists are needed to define a genre adequately, the author conducted some experiments using different training set sizes. In Figure 5.3, the results of these experiments for 9-NN classification using the combinations AIT+M and AIT+MR are depicted. The percentage of classified artists is 100% for all combinations shown. It can be seen that t_{15} and t_8 provide high accuracies of up to 85% and 78%, respectively. Examining the results of the t_4 and t_2 settings reveals much lower accuracies. The impact of the training set size on classification performance was also investigated for the TF·IDF approach. In this case, the results do not depend on an additional parameter z. Therefore, the classification accuracies for different t_x experiments are summarized in Table 5.6, rather than in a plot. From the table, it can be seen that the MR scheme, in general, performs better than the MGS scheme. Furthermore, comparing the accuracies achieved with co-occurrence analysis to those obtained with TF·IDF, co-occurrence analysis performs remarkably worse than TF·IDF, especially for low x values of t_x. For t_4, co-occurrence analysis yields a maximum of 61% accuracy (AIT+MR), whereas TF·IDF yields 68% (MR). For t_2, the results differ even more. The co-occurrence approach yields a maximum of 35% (AIT+MR) for this setting, whereas the TF·IDF approach achieves 43% (MR). To conclude, the additional terms incorporated by the TF·IDF approach are highly valuable when the similarity model is based only on a few Web pages, a scenario which was emulated by the t_x experiments with low values of x.

As for the comparison of the co-occurrence approach with the TF·IDF approach on collection *C103a22g*, Tables 5.7 and 5.8 summarize the results. In these tables, accuracies for different k-NN classification experiments and different query schemes are depicted. The threshold for the confidence filter was set to $z = 0$. Therefore, it is possible to distinguish between the accuracies achieved for the artists that have been classified (column *accuracy*$_{cls}$ and those achieved when treating artists with a lack of available information as misclassified (column *accuracy*$_{all}$). Column *classified* gives the percentage of artists for

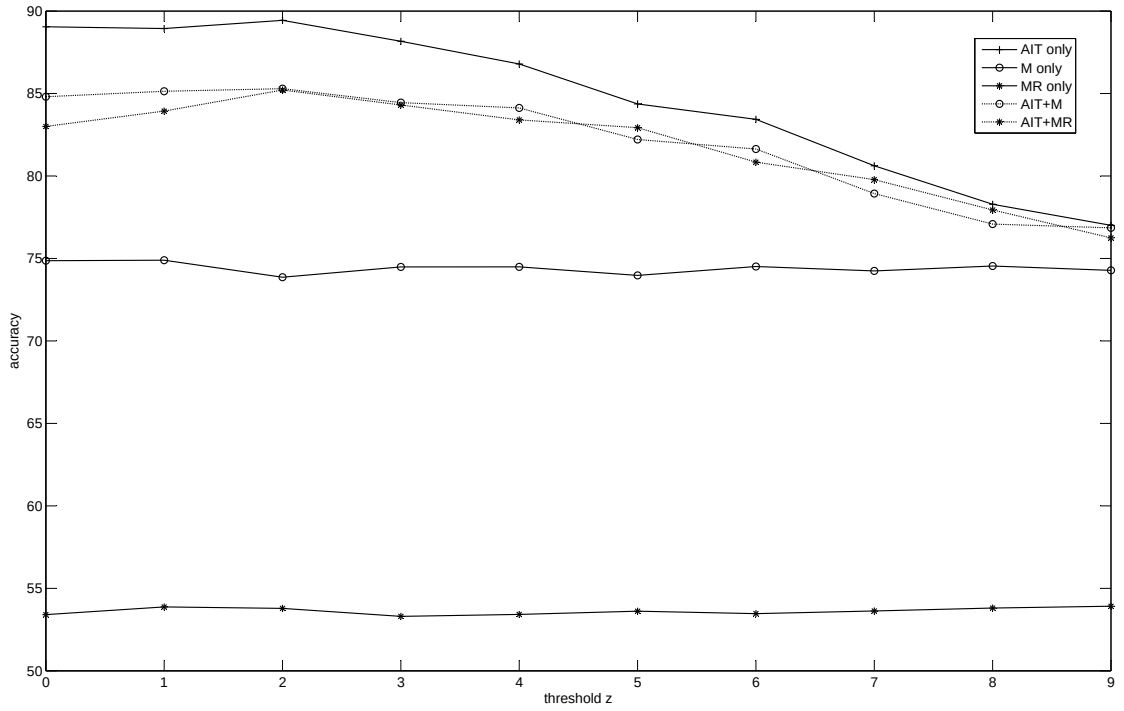


Figure 5.1: Accuracies, in percent, for single and compound similarity measures using 9-NN t_{15} validation and the confidence filter on *C224a14g*.

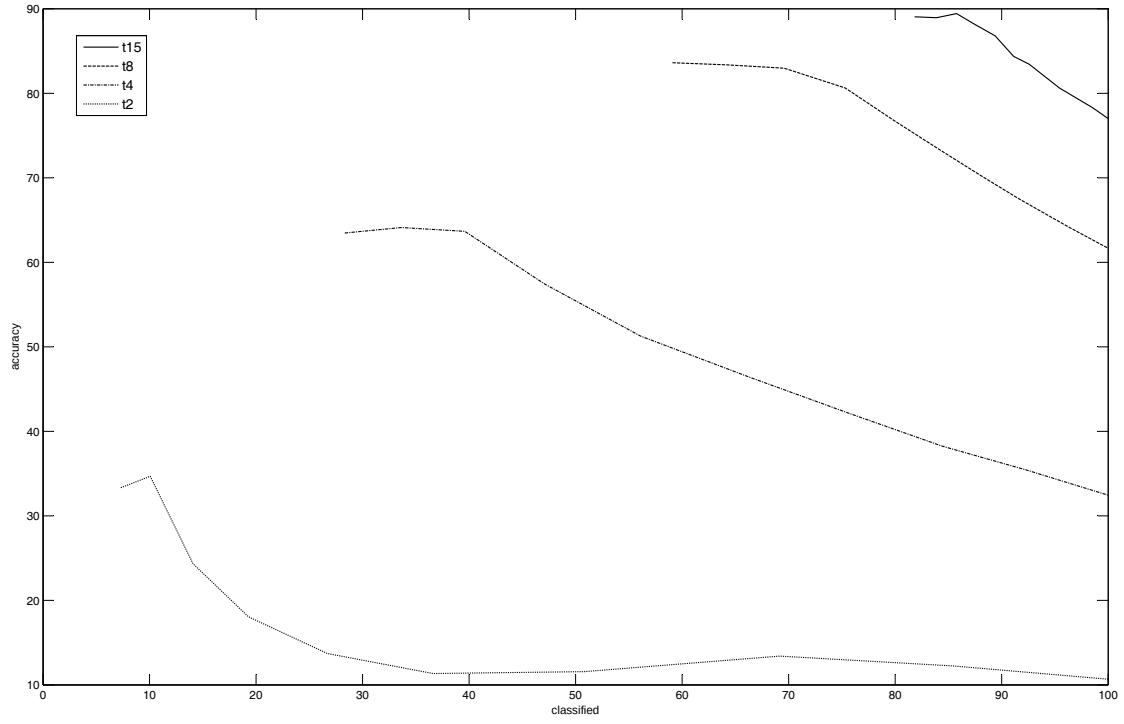


Figure 5.2: Accuracy for the single AIT measure, plotted against percentage of classified artists for different training set sizes and 9-NN classification on *C224a14g*.

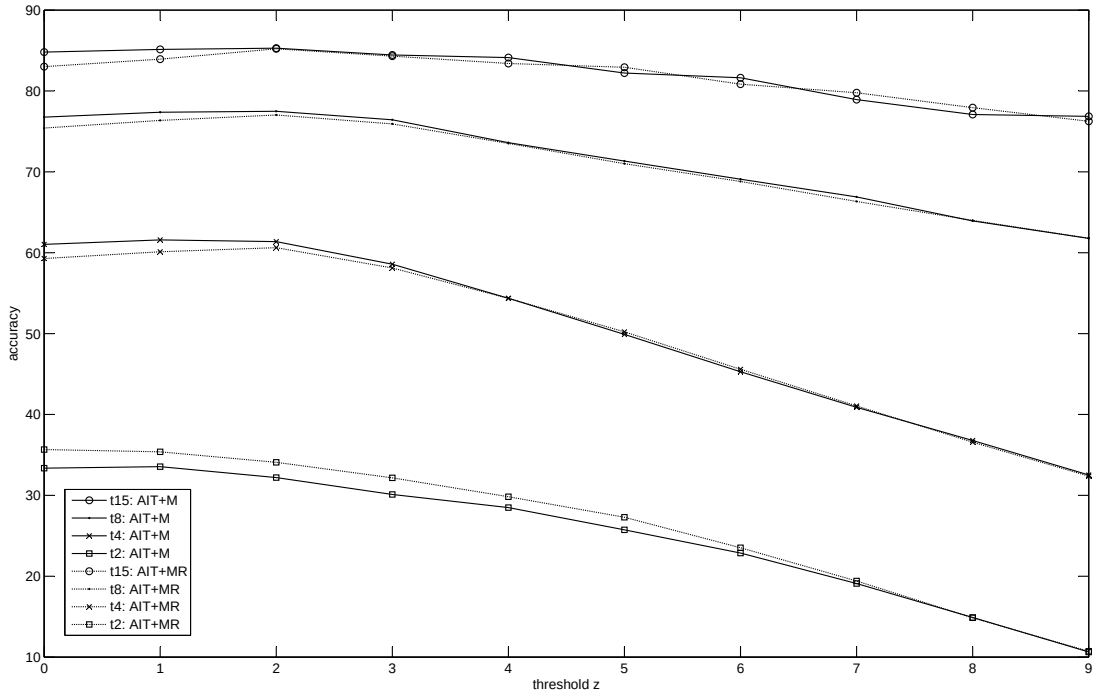


Figure 5.3: Accuracies for 9-NN classification experiments, in percent, for different combinations of the query schemes AIT, M, and MR and different training set sizes on *C224a14g*.

which enough information was available to perform classification.

In Table 5.7, the results for the co-occurrence approach using the confidence filter is depicted for 1-NN and 3-NN classification experiments with *leave-one-out cross-validation*. The upper four rows of this table show the results using single similarity measures according to the M, MR, MGS, and AIT schemes. It can be seen that AIT performs best (nearly 80% accuracy using 3-NN classification), while the M setting performs worst (only 36% accuracy using 3-NN). However, the high accuracies of AIT come along with the drawback of many unclassified data items. Indeed, taking a closer look at column *accuracy*$_{all}$), where unclassified data items are treated as errors, reveals that the MR and MGS schemes are much more accurate than AIT if the goal is to make a prediction for as many artists as possible. This can be explained by the fact that the probability for two artist names to appear in the title of a Web page is much lower than to co-occur anywhere inside a page. Comparing the MR and MGS settings shows largely equal performances with slight advantages for MGS.

The lower three rows of Table 5.7 depict the evaluation results for combining the AIT measure with the measure derived from the other schemes and taking the confidence filter as decision guidance. More precisely, AIT is used for those artists which co-occur with at least k other artists (k according to k-NN). Already one single data item with a similarity of zero among the k nearest neighbors causes a fallback to one of the other similarity measures. It can be seen that using the compound measures

considerably raises the percentage of classified artists, while at the same time accuracies decrease only by a small amount. In fact, accuracies of nearly 70% can be achieved at up to 98% classified artists using 3-NN classification with MR or MGS as second similarity measure. The reason for not being able to classify all data items, even after applying the second similarity measure, is the lack of data for the artists *Stan Getz, João & Astrud Gilberto* and *The Heavenly Light Quartet*. While for the first, no Web pages could be found, the second only co-occurred with the artist *Heavenly* on the four Web pages found for *The Heavenly Light Quartet*.

Table 5.8 shows the k-NN classification results for the TF·IDF approach using leave-one-out cross-validation. In general, MGS performs slightly better than MR. The highest accuracy is about 68% for MGS with 3-NN classification. Querying with the MR scheme, Web pages for all artists were available, whereas the MGS scheme did not return pages for *Stan Getz, João & Astrud Gilberto*.

To investigate which genres are likely to be confused with others, confusion matrices for the best performing experimental settings are depicted in Figure 5.4 for the co-occurrence approach and in Figure 5.5 for the TF·IDF approach. Unclassified artists are treated as errors and are omitted in the confusion matrices. The diagonal elements represent the accuracies, in percent. The genres "Blues", "Celtic", "Euro-Dance", and "Folk-Rock" are perfectly distinguished by both techniques. Also "DnB" shows high accuracies of at least 80% for both approaches. In contrast, the results for the genres "A Cappella", "Electronic", and "Jazz" are at the lower end of the performance scale for both approaches. The frequent misclassification of "Jazz" as "Jazz Guitar" is comprehensible to some extent. More interesting than the genres for which both techniques behave similarly are those with differing results. Comparing Figures 5.4 and 5.5, it becomes obvious that TF·IDF performs much better than co-occurrence analysis for the genre "Death Metal". On the other hand, the latter shows better results for the electronic genres "Downtempo", "Trance", and "Trance2". For the other genres, no significant differences are apparent.

	1-NN			3-NN		
	$accuracy_{cls}$	*classified*	$accuracy_{all}$	$accuracy_{cls}$	*classified*	$accuracy_{all}$
M only	35.30	99.03	34.95	36.27	99.03	35.93
MR only	52.94	99.03	52.43	53.47	98.06	52.43
MGS only	57.43	98.06	56.31	53.54	96.12	51.46
AIT only	67.09	76.70	51.46	79.25	51.46	40.78
AIT+M	62.75	99.03	62.14	62.75	99.03	62.14
AIT+MR	62.75	99.03	62.14	69.31	98.06	67.96
AIT+MGS	64.36	98.06	63.11	69.70	96.12	66.99

Table 5.7: Accuracies, in percent, for k-NN evaluations using co-occurrence analysis on collection *C103a22g*.

	1-NN	3-NN	7-NN	
	$accuracy_{all}$			*classified*
MR	62.14	58.25	57.28	100.00
MGS	66.02	67.96	60.19	99.03

Table 5.8: Accuracies, in percent, for k-NN evaluations using the TF·IDF approach on collection *C103ag22*.

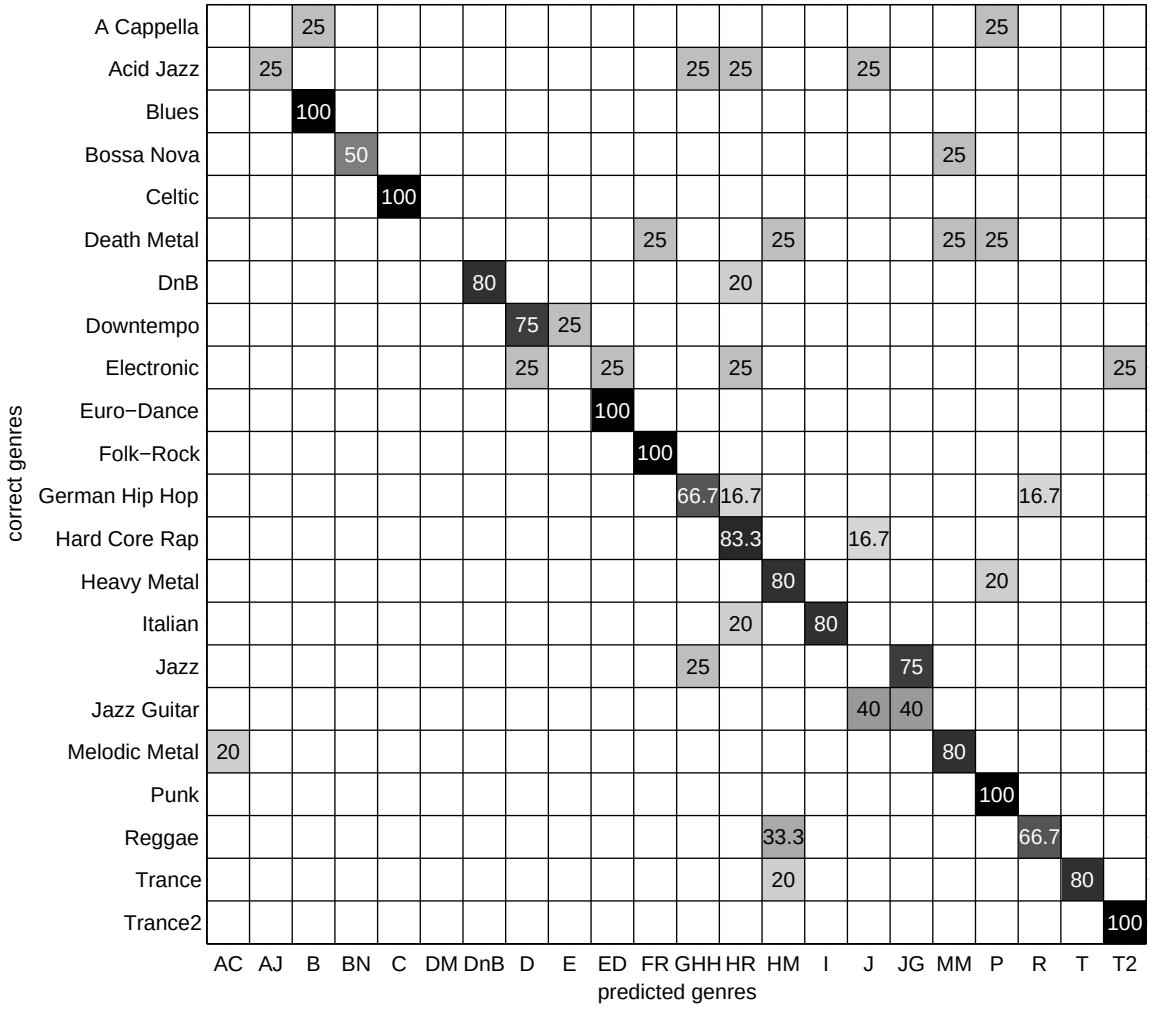


Figure 5.4: Confusion matrix for the co-occurrence approach using 3-NN classification and the compound AIT+MGS measure on *C103a22g*.

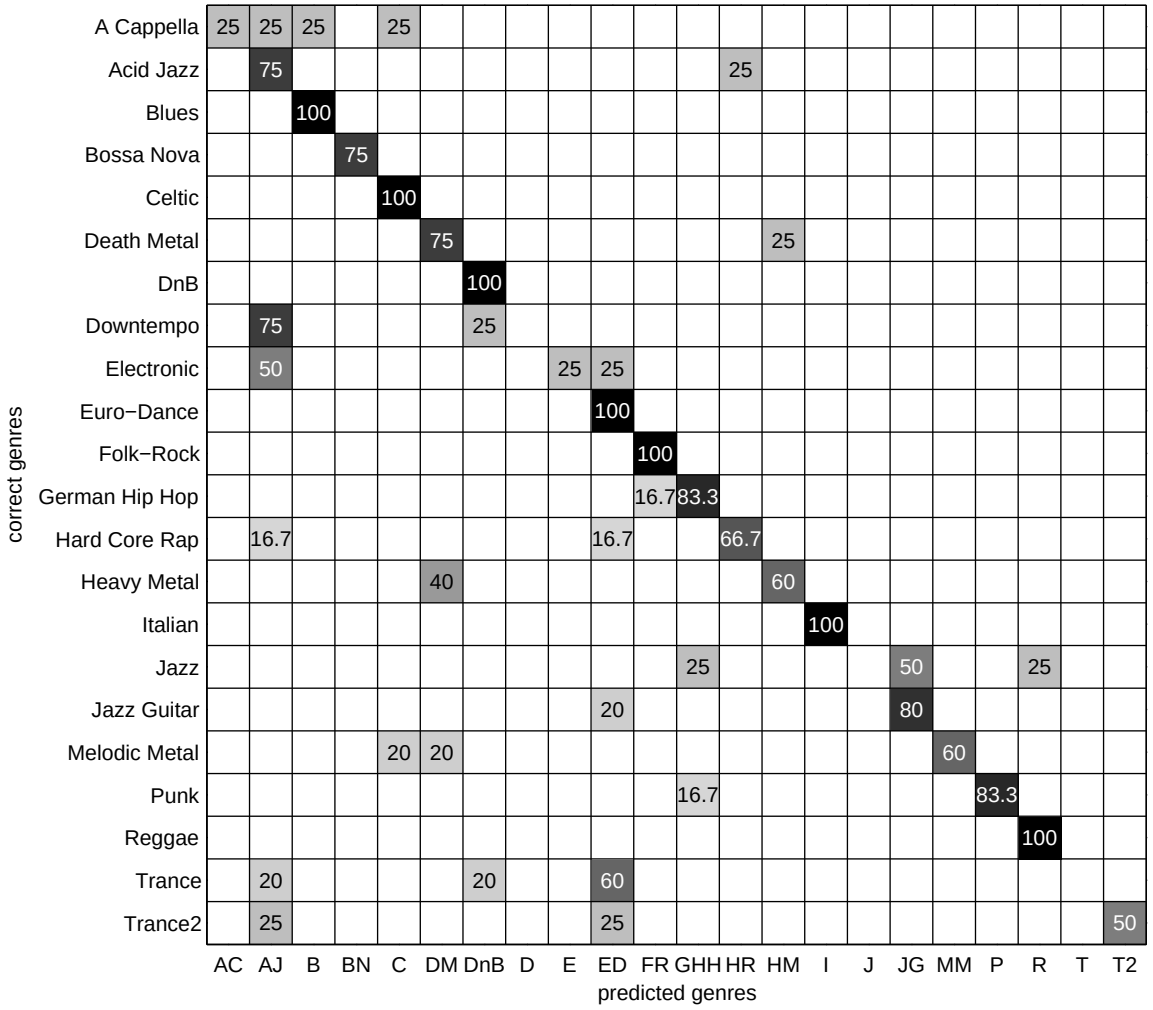


Figure 5.5: Confusion matrix for the TF·IDF approach using 3-NN classification and the MGS scheme on *C103a22g*.

5.3.2 Artist Prototypicality

To analyze the different approaches to prototypical artist ranking, which are the *BL/FL model* according to Subsection 3.3.2, the *BL/FL model with penalization of exorbitant popularity* according to Subsection 3.3.2, and a third model derived from the page counts for queries of the form "*artist name*"+"*genre name*", similar to the method described in Subsection 3.3.1. This third model will be called *page counts model* in the following. It ranks the artists within each genre according to their page counts, thus yielding a popularity ranking. Since prototypicality is strongly related to popularity, it is reasonable to use this model for prototypicality estimation.

Evaluating the quality of the prototypicality ranking approaches is a difficult task for various reasons. First, prototypicality is influenced by personal taste and cultural context. Thus, if a number of different people are asked which artists they consider prototypical for a certain genre, they would most probably name their favorites or artists from their own country of origin. Another issue is that prototypical artists may also change over time. For example, formerly unknown artists may become very popular overnight. Since evaluation should be performed on a large artist set, conducting a user study to obtain a ground truth was out of the question as this would have included ranking every artist with respect to all other artists of the respective genre. Alternatively, presenting only a subset of artists would have resulted in incomplete rankings. The choice for using the collection *C1995a9g* as ground truth was motivated by the fact that it is a subset of the collection on which AGMIS is based, cf. Section 6.1. More precisely, *C1995a9g* comprises artists from 9 genres, for which information on tiers are available on AMG. The artists of each genre are usually clustered in three tiers according to their importance for the respective genre, which is defined by music experts.

The quality of the prototypicality rankings given by the three models under evaluation was assessed via *accuracy estimation on a classification task* and calculating *Spearman's rank-order correlation* between the rankings given by the models and the rankings given by AMG.

Classification Accuracy

To gain an overall impression of the performance of the investigated approaches, the AMG tiers were interpreted as classes, and a classification task was simulated using the prototypicality ratings as classifiers. To this end, the author mapped the rankings obtained by the prototypicality estimation approaches to the ones given by the AMG tiers and determined concordances. More precisely, given that the prototypicality algorithm under evaluation has produced a specific ranking R of the artists of a genre and assuming the three AMG tiers for this genre contain n_1, n_2, and n_3 artists, respectively, the first n_1 elements of R are assigned to tier 1, the next n_2 to tier 2, and the last n_3 to tier 3. These assignments can then be regarded as classification decisions, and classification accuracy values can be computed.

Results and Discussion

The overall results of the classification task are depicted in Figure 5.6, where a confusion matrix for each of the three investigated approaches is shown. The columns indicate the tiers to which the approaches map their rankings, the rows indicate the actual AMG tiers. Accuracy values are given in percent. It can be seen that the BL/FL models, in general, perform better than the page counts approach, especially for predicting 1st-tier-artists. Comparing the BL/FL model without penalization to the BL/FL model with penalization of exorbitant popularity reveals slightly better results for the latter when predicting 1st-tier-artists, but slightly worse results for predicting tiers 2 and 3. This becomes particularly obvious when considering Table 5.9, where the top-ranked artists for the genres "Heavy Metal" and "Folk" are shown. This table reveals the impact of the penalization on artists whose names equal common speech words, especially for the heavy metal artists. In fact, the BL/FL model and the page counts approach highly rank artists like *Death*, *Europe*, *Tool*, *Kiss*, and *Filter*. The same artists are considerably downranked by the BL/FL approach with penalization. The rankings for the genre "Folk", in contrast, remain almost unmodified as the artists of this genre are usually known by their real name. To investigate the impact of the genre on the quality of the results, Figure 5.7 shows confusion matrices for each of the 9 genres for the best performing BL/FL approach with penalization. It can be seen that the overall results for the genre "Electronica" are by far the best. Weighted with the number of artists in every tier, accuracy values of 83% could be obtained for this genre, which is 11 percentage points above the baseline of 72% given by tier 2, cf. Table A-8. The remarkable wrong confusion for correct tier 3 in genre "Folk" is caused by one single artist of tier 3, which is incorrectly classified as belonging to tier 1. Hence, this misclassification does not considerably influence the overall performance of the approach.

Table 5.10 shows the overall genre-specific accuracies for the three prototypicality models, obtained by weighting the genre-specific accuracies, cf. Figure 5.7, with the number of artists in every tier. acc_0 denotes the accuracy for the evaluated ranking approach to map an artist exactly to the same AMG tier it should fall into according to AMG's ranking. acc_1 denotes the accuracy when deviations of up to one tier are allowed. Comparing Table 5.10 to Table A-8, which gives the baseline for the classification experiments, shows that the overall acc_0 accuracies, except those for the genre "Rap", considerably exceed the baseline. In the case of "Electronica", "Reggae", "Jazz", and "RnB", they are even between 10% and more than 20% above the baseline. In contrast, the results for the genre "Rap" are rather poor. Taking a closer look at AMG's assignment of rap artists to tiers reveals that this may be caused by a subjective and time-dependent bias of AMG's editors. Indeed, very popular rap artists like *Eminem* and *Snoop Dogg* are assigned to the 2nd tier, whereas many artists that were very popular some years ago are still assigned to tier 1.

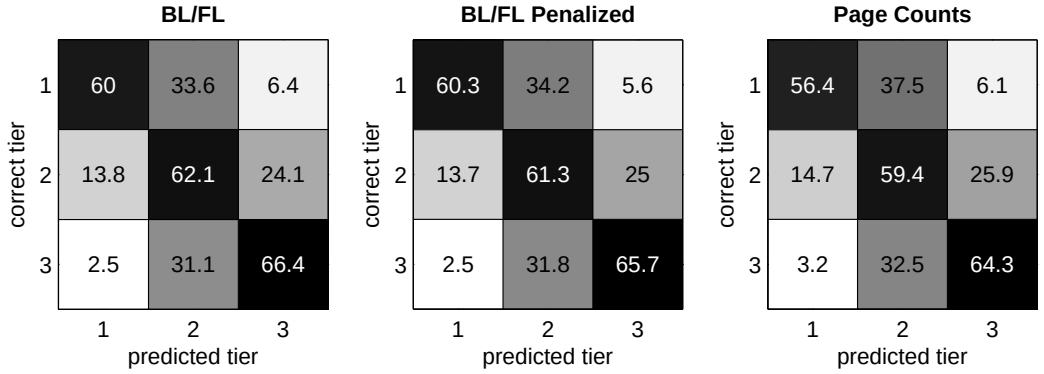


Figure 5.6: Confusion matrices of the classification task for each of the three prototypicality ranking approaches.

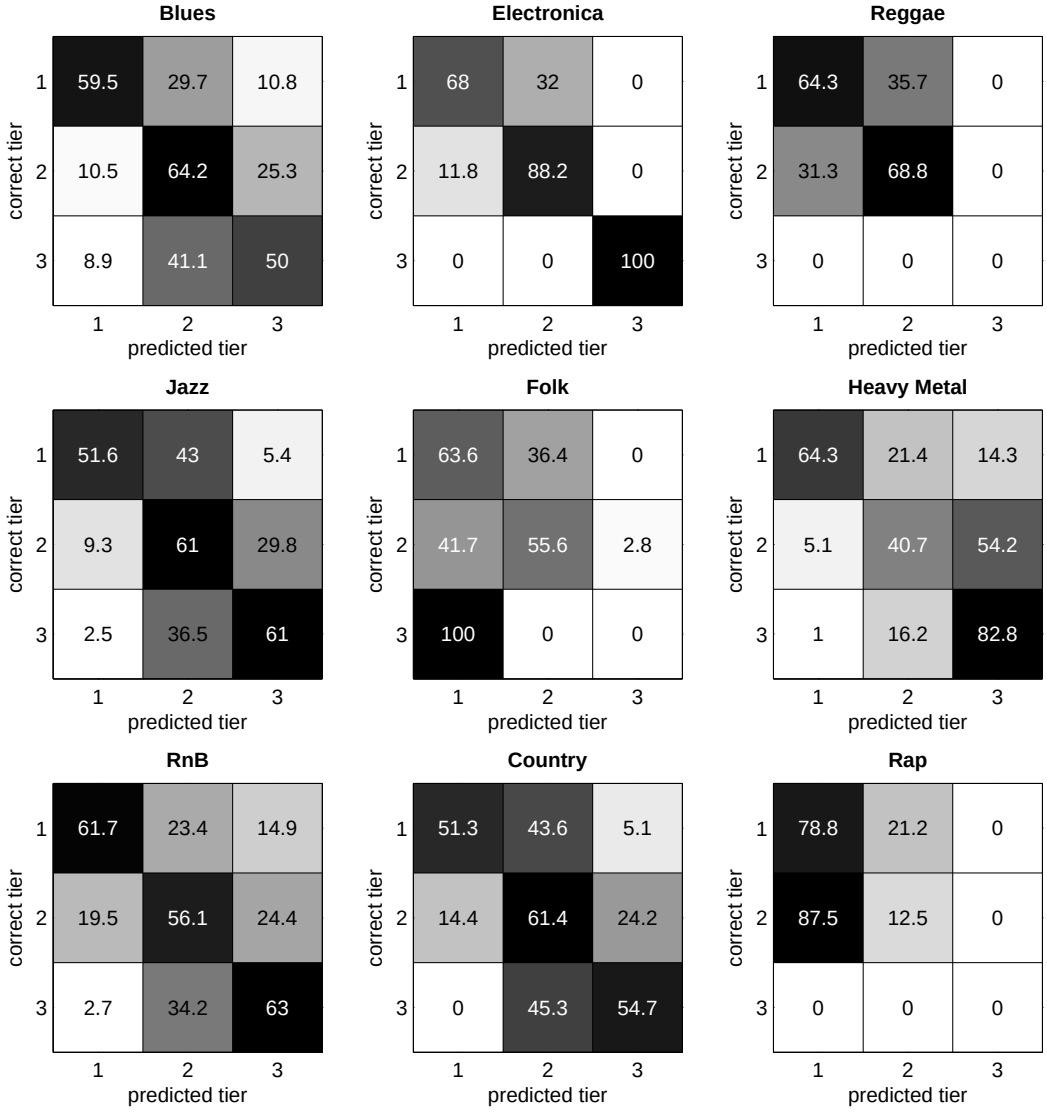


Figure 5.7: Confusion matrices of the classification task using the BL/FL approach with penalization of exorbitant popularity, shown for every genre.

Heavy Metal		
BL/FL	BL/FL Penalized	Page Counts
Death	Metallica	Metallica
Europe	AC/DC	Death
Tool	Black Sabbath	Kiss
Metallica	Death	Tool
Kiss	Led Zeppelin	Extreme
Filter	Riot	Europe
AC/DC	Iron Maiden	Trouble
Led Zeppelin	Judas Priest	Iron Maiden
Black Sabbath	Slayer	Filter
Alice Cooper	Marilyn Manson	Rainbow

Folk		
BL/FL	BL/FL Penalized	Page Counts
Woody Guthrie	Woody Guthrie	Woody Guthrie
Joan Baez	Joan Baez	Joan Baez
Lucinda Williams	Judy Collins	Pete Seeger
Pete Seeger	Pete Seeger	Lucinda Williams
Judy Collins	Lucinda Williams	Arlo Guthrie
Leadbelly	Doc Watson	Doc Watson
Doc Watson	Leadbelly	Judy Collins
Townes Van Zandt	Phil Ochs	Alan Lomax
Gordon Lightfoot	Gordon Lightfoot	Leadbelly
Phil Ochs	Townes Van Zandt	Gordon Lightfoot

Table 5.9: The 10 top-ranked artists of the genres "Heavy Metal" and "Folk" for each of the three prototypicality models.

	BL/FL		BL/FL Pen		Page Counts	
Genre	acc_0	acc_1	acc_0	acc_1	acc_0	acc_1
Blues	0.59	0.95	0.59	0.95	0.57	0.94
Electronica	0.81	1.00	0.83	1.00	0.73	0.98
Reggae	0.67	1.00	0.67	1.00	0.70	1.00
Jazz	0.60	0.98	0.60	0.98	0.60	0.99
Folk	0.62	0.99	0.60	0.99	0.62	0.99
Heavy Metal	0.73	0.98	0.73	0.99	0.67	0.98
RnB	0.62	0.95	0.60	0.96	0.50	0.93
Country	0.59	0.99	0.58	0.99	0.59	0.99
Rap	0.66	1.00	0.66	1.00	0.66	1.00

Table 5.10: Overall genre-specific accuracies for the three prototypicality models.

Spearman's Rank-Order Correlation

To measure the correlation between the ground truth ranking given by AMG and the rankings obtained with the prototypicality models, the author used the well established *Spearman's rank-order correlation coefficient*, e.g., [Sheskin, 2004]. Since the rankings by AMG are strongly tied, using the standard formulation would spuriously inflate the correlation values. Therefore, the tie-corrected variant given in Formulas 5.3–5.6 was employed. In these formulas, r_{S_c} gives the rank-order correlation coefficient, n is the total number of ranked data items, X and Y represent the two rankings under consideration, s_X and s_Y are the numbers of sets of ties present in X and Y, respectively, and t_{X_i} and t_{Y_i} are the numbers of X and Y scores that are tied for a given rank.

$$r_{S_c} = \frac{\sum x^2 + \sum y^2 - \sum d^2}{2 \cdot \sqrt{\sum x^2 \cdot \sum y^2}} \tag{5.3}$$

$$\sum x^2 = \frac{n^3 - n - T_X}{12} \qquad \sum y^2 = \frac{n^3 - n - T_Y}{12} \tag{5.4}$$

$$T_X = \sum_{i=1}^{s_X} (t_{X_i}^3 - t_{X_i}) \qquad T_Y = \sum_{i=1}^{s_Y} (t_{Y_i}^3 - t_{Y_i}) \tag{5.5}$$

$$\sum d^2 = \sum_{i=1}^{n} (X_i - Y_i)^2 \tag{5.6}$$

Results and Discussion

As for the results of the correlation analysis, in Table 5.11, the Spearman's rank-order correlations between the ground truth ranking and the rankings obtained with the prototypicality models are shown for each genre. Except for the genre "Rap", the rank-order correlation coefficient is at least 0.3. For the genres "Electronica", "Jazz", "Heavy Metal", and "RnB", it is about 0.5, and for "Country" it almost reaches 0.6. Significance tests proved significance for all obtained correlations, except those for the genre "Rap". For this genre, the weak negative correlation was not stated significant.

5.3.3 Descriptive Artist Properties

As for the task of retrieving terms that represent descriptive properties of a given artist, the author of this PhD thesis conducted a user study to assess different term weighting functions with respect to their suitability to find such descriptive terms, cf. Subsection 2.2.1. The corpus of Web pages used for evaluation was generated by fetching the 100 top-ranked pages returned for queries to Google using the MR scheme.

Genre	BL/FL	BL/FL Pen	Page Counts
Blues	0.40	0.39	0.35
Electronica	0.45	0.48	0.37
Reggae	0.31	0.30	0.31
Jazz	0.49	0.49	0.53
Folk	0.31	0.34	0.33
Heavy Metal	0.48	0.48	0.41
RnB	0.55	0.55	0.42
Country	0.58	0.59	0.58
Rap	-0.21	-0.21	-0.15
Mean	**0.37**	**0.38**	**0.35**

Table 5.11: Spearman's rank-order correlations between the ground truth ranking by AMG and the rankings obtained with the prototypicality ranking approaches.

Term Weighting Functions

To investigate the quality of the term weighting functions document frequency, term frequency, and TF·IDF, cf. Subsection 3.3.3, for determining descriptive artist-related terms, the author conducted a user study. The choice of the term weighting function is crucial not only for the presentation of descriptive terms on artist pages provided by AGMIS, but also for the Sunburst creation step of the COB, cf. Subsection 4.2.6, as it influences the quality of the hierarchical clustering, the hierarchical layout, and thus the visualization of the COB.

For this study, a subset of the collection *C224a14g* was used. Taking only 8 instead of 16 artists per genre yielded the used collection *C112a14g*. This artist selection was performed by the author's judgment of artist popularity. The dictionary used for indexing contained 1,506 musically relevant terms, cf. Subsection 3.2. To generate the data for evaluation, the 10 most important terms according to each of the three term weighting functions were calculated on the Web pages retrieved for each artist. To avoid biasing of the results, for each artist, the 10 terms obtained by applying every weighting function were then merged. Hence, each participant was presented a list of 112 artist names and, for each name, a set of associated terms, i.e., a mixture of the terms obtained by the three weighting functions. Since the author had no a priori knowledge of which artists were known by which participant, the participants were told to evaluate only those artists they were familiar with. Their task was then to rate the associated terms with respect to their appropriateness for describing the artist or his/her music. To this end, they had to assign each term to one of the three classes $+$ *(good description)*, $-$ *(bad description)*, and $\sim$ *(indifferent or not wrong, but not a very discriminative description of the artist)*. The number of participants in the user study was five. Three of them were computer science students, the other two researchers in computer science. All of them were male, and all stated to listen to music often.

Artist	Assessments	TF	DF	TF·IDF	TF_{avg}	DF_{avg}	$TF{\cdot}IDF_{avg}$
50 Cent	3	17	16	19	5.67	5.33	6.33
ABBA	3	10	11	5	3.33	3.67	1.67
Al Green	1	-2	0	-4	-2.00	0.00	-4.00
Alice Cooper	3	8	5	1	2.67	1.67	0.33
Alice in Chains	2	10	12	7	5.00	6.00	3.50
Alpha Blondie	1	-10	-8	-8	-10.00	-8.00	-8.00
Anthrax	2	6	9	5	3.00	4.50	2.50
Antonin Dvorak	2	6	9	9	3.00	4.50	4.50
Aphex Twin	2	13	13	9	6.50	6.50	4.50
Aretha Franklin	3	9	8	9	3.00	2.67	3.00
Bad Religion	3	4	17	8	1.33	5.67	2.67
Basement Jaxx	1	7	8	7	7.00	8.00	7.00
BB King	3	-1	0	-1	-0.33	0.00	-0.33
Beck	3	-4	-6	0	-1.33	-2.00	0.00
Belle and Sebastian	2	-1	-3	-2	-0.50	-1.50	-1.00
Big Bill Broonzy	1	4	4	3	4.00	4.00	3.00
Billie Holiday	2	9	8	7	4.50	4.00	3.50
Black Sabbath	3	10	10	11	3.33	3.33	3.67
Bob Dylan	3	4	8	10	1.33	2.67	3.33
Bob Marley	3	-5	-3	1	-1.67	-1.00	0.33
Britney Spears	3	10	18	15	3.33	6.00	5.00
Carl Cox	1	8	7	8	8.00	7.00	8.00
Chemical Brothers	3	5	8	6	1.67	2.67	2.00
Chuck Berry	1	1	1	3	1.00	1.00	3.00
Cypress Hill	2	6	2	6	3.00	1.00	3.00
Daft Punk	2	6	9	3	3.00	4.50	1.50
Dave Brubeck	2	5	4	1	2.50	2.00	0.50
Dead Kennedys	1	5	6	4	5.00	6.00	4.00
Deep Purple	3	6	7	3	2.00	2.33	1.00
Dixie Chicks	1	6	5	6	6.00	5.00	6.00
Django Reinhardt	2	9	9	8	4.50	4.50	4.00
Dolly Parton	1	4	4	1	4.00	4.00	1.00
Dr. Dre	2	11	12	3	5.50	6.00	1.50
Duke Ellington	3	11	10	5	3.67	3.33	1.67
Elvis Presley	4	-3	-4	-5	-0.75	-1.00	-1.25
Eminem	4	22	15	15	5.50	3.75	3.75
Faith Hill	1	4	4	2	4.00	4.00	2.00
Fatboy Slim	2	5	6	1	2.50	3.00	0.50
Frederic Chopin	3	4	-1	0	1.33	-0.33	0.00
Garth Brooks	1	3	3	2	3.00	3.00	2.00
Glenn Miller	1	0	0	0	0.00	0.00	0.00
Grandmaster Flash	1	1	3	3	1.00	3.00	3.00
Hank Williams	1	4	3	2	4.00	3.00	2.00
Howlin' Wolf	1	1	1	-2	1.00	1.00	-2.00
Iron Maiden	3	10	11	11	3.33	3.67	3.67

Table 5.12: Results of the user study on different term weighting functions.

Artist	Assessments	TF	DF	TF·IDF	TF_{avg}	DF_{avg}	$TF{\cdot}IDF_{avg}$
James Brown	2	-1	1	-1	-0.50	0.50	-0.50
Janet Jackson	2	3	5	1	1.50	2.50	0.50
Jimmy Cliff	1	-1	-2	1	-1.00	-2.00	1.00
Joan Baez	1	7	7	5	7.00	7.00	5.00
Johann Sebastian Bach	1	4	4	4	4.00	4.00	4.00
Johannes Brahms	2	11	11	11	5.50	5.50	5.50
John Lee Hooker	1	0	0	2	0.00	0.00	2.00
John Mayall	1	-1	-1	-3	-1.00	-1.00	-3.00
Johnny Cash	2	11	11	7	5.50	5.50	3.50
Justin Timberlake	3	-2	0	-2	-0.67	0.00	-0.67
Kraftwerk	1	6	4	2	6.00	4.00	2.00
Little Richard	2	-3	-1	-3	-1.50	-0.50	-1.50
Louis Armstrong	2	-3	-4	-3	-1.50	-2.00	-1.50
Ludwig van Beethoven	1	5	6	1	5.00	6.00	1.00
Madonna	3	13	6	7	4.33	2.00	2.33
Marvin Gaye	1	3	4	0	3.00	4.00	0.00
Megadeth	1	0	3	-2	0.00	3.00	-2.00
Michael Jackson	2	-9	-9	-10	-4.50	-4.50	-5.00
Miles Davis	1	-2	-3	0	-2.00	-3.00	0.00
Missy Elliot	2	9	11	11	4.50	5.50	5.50
Moloko	2	11	9	7	5.50	4.50	3.50
Muddy Waters	1	0	-2	-2	0.00	-2.00	-2.00
N'Sync	4	5	6	4	1.25	1.50	1.00
Nirvana	1	1	0	3	1.00	0.00	3.00
NoFX	2	15	15	-6	7.50	7.50	-3.00
Patti Smith	1	1	4	4	1.00	4.00	4.00
Prince	2	-1	-1	1	-0.50	-0.50	0.50
Public Enemy	2	10	12	7	5.00	6.00	3.50
Radiohead	1	6	6	6	6.00	6.00	6.00
Ramones	1	3	6	-1	3.00	6.00	-1.00
Run DMC	3	9	1	1	3.00	0.33	0.33
Sepultura	2	11	5	4	5.50	2.50	2.00
Sex Pistols	2	6	8	4	3.00	4.00	2.00
Shaggy	2	3	-2	3	1.50	-1.00	1.50
Sid Vicious	1	-1	1	1	-1.00	1.00	1.00
Slayer	1	-2	0	-3	-2.00	0.00	-3.00
Smashing Pumpkins	2	-2	-2	-2	-1.00	-1.00	-1.00
Solomon Burke	1	2	2	3	2.00	2.00	3.00
Sonic Youth	1	4	7	5	4.00	7.00	5.00
Suzanne Vega	2	4	6	2	2.00	3.00	1.00
The Animals	1	-4	-4	-4	-4.00	-4.00	-4.00
The Clash	1	2	0	-2	2.00	0.00	-2.00
The Kinks	1	1	0	1	1.00	0.00	1.00
The Rolling Stones	4	-1	5	-3	-0.25	1.25	-0.75
Tracy Chapman	1	2	4	1	2.00	4.00	1.00
Wolfgang Amadeus Mozart	2	12	12	8	6.00	6.00	4.00
Ziggy Marley	1	1	1	4	1.00	1.00	4.00
Sum	**172**	**386**	**413**	**271**	**204.08**	**224.00**	**141.08**

Table 5.13: Continuation of Table 5.12.

Results and Discussion

The author received a total of 172 assessments for sets of terms assigned to a specific artist. 92 out of the 112 artists were covered. To analyze the results, for each artist and for each weighting function, the sum of all points obtained by the individual assessments was calculated. As for the mapping of classes to points, each term in class $+$ contributes 1 point, each term in class $-$ gives -1 point, and each term in class $\sim$ yields 0 points. Summing up the points over all assessments of each artist, for the three term weighting functions, gives the results shown in the columns labeled *TF*, *DF*, and *TF·IDF* of Tables 5.12 and 5.13. Only the 92 artists that were assessed at least once are depicted. The column labeled *Assessments* shows the number of assessments made, i.e., the number of test persons who evaluated the respective artist. The next three columns reveal, for each weighting function, the summed up ratings over all terms, in points. Since the performance of the term weighting functions is hardly comparable between different artists using the summed up points, columns TF_{avg}, DF_{avg}, and $TF{\cdot}IDF_{avg}$ illustrate the averaged scores, which are obtained by dividing the summed up points by the number of assessments. These averaged points reveal that the quality of the terms vary strongly between different artists. Nevertheless, it can be stated that, for most artists, the number of descriptive terms exceeds the number of the non-descriptive ones. To investigate the overall performance of the term weighting functions, the arithmetic mean of the averaged points over all artists were calculated. These were 2.22, 2.43, and 1.53 for TF, DF, and TF·IDF, respectively. Due to the performed mapping from classes to points, these values can be regarded as the average excess of the number of good terms over the number of bad terms. Hence, overall, the document frequency measure performed best, the term frequency second best, and the TF·IDF worst for this specific task of finding descriptive terms for a music artist based on a dictionary of musically relevant terms.

To investigate the significance of these results, the author performed *Friedman's non-parametric two-way analysis of variance*, cf. [Friedman, 1940] or [Sheskin, 2004]. This test is similar to the two-way ANOVA, but does not assume a normal distribution of the data. The test showed that the variance differences in the results seem to be significant with a very high probability. Moreover, pairwise comparisons between the results given by the three term weighting functions showed that TF·IDF performed significantly worse than both TF and DF, whereas no significant difference could be made out between the results obtained using DF and those obtained using TF.

The laborious task of combining and analyzing the different assessments of the participants in the user study further allowed the author to take a qualitative look at the terms. Although the majority of the terms was judged descriptive, some interesting flaws were discovered. First, the term "musical" occurred on quite a lot of Web pages and was therefore often contained in the set of the top-ranked terms. However, no participant judged this term as descriptive for any artist. A similar observation

could be made for the term "real". In this case, however, one participant stated that this is a term commonly used in the context of hip-hop music and may therefore be descriptive to some extent. Furthermore, the term "christmas" was associated occasionally to some artists. These associations seem quite random since none of the artists is known for his/her performance of Christmas carols. Another reason for erroneously assigning a term to an artist is terms that are part of artist, album, or song names, but are not suited well to describe the respective artist. Examples for this problem category are "infinite" for the band *Smashing Pumpkins* and "human" as well as "punk" for the band *Daft Punk*.

5.3.4 Band Members and Instrumentation

The approach for determining band members and their instruments proposed in Subsection 3.3.4 was assessed using the test collections *C51a240m* and *C51a499m*, which were compiled from the author's private music repository. Defining the ground truth, i.e., the actual band members and instruments, is a labor-intensive and time-consuming task. Therefore, the data set was restricted to 51 bands, with a strong focus on the genre "Metal". The chosen bands vary strongly with respect to their popularity. Some are very well known, like *Metallica*, but most are largely unknown, like *Powergod*, *Pink Cream 69*, or *Regicide*. The current line-up of the bands was gathered by consulting *Wikipedia* [wik, 2007b], *AMG* [amg, 2007a], *Discogs* [dis, 2007], or the band's Web site. Eventually, a total number of 240 current members with their respective instruments were made out for the 51 bands, yielding collection *C51a240m*. Since the author further aimed at investigating the performance of the approach on the task of finding members that already left the band, all former band members were sought using the same sources as for the creation of *C51a240m*. This second ground truth data set contains 499 band members.

The author conducted different evaluation experiments to assess the quality of the approach proposed in Subsection 3.3.4. First, *precision* and *recall* of the predicted (member, instrument)-pairs on the ground truth were calculated using a fixed filtering threshold t_{DF} in the member and instrument prediction step, cf. Subsection 3.3.4. To get an impression of the goodness of the recall values, the author also determined the *upper bound for the recall* achievable with the proposed method. Such an upper bound does exist since predicting any band member is obviously only possible if the member's name actually occurs in at least one Web page retrieved for the band under consideration. In a subsequent experiment, the influence of the parameter t_{DF} on precision and recall was investigated. Each evaluation experiment was conducted on both collections *C51a240m* and *C51a499m* using each of the four query schemes M, MR, MM, and LUM to retrieve up to 100 top-ranked Web pages returned by Google.

Three different string comparison methods were employed for evaluation. First, *exact string matching*

was used. Accounting for the problem of different spelling for the same artist[3], the approach was also evaluated on the basis of a *canonical representation* of each band member. To this end, the author performed a mapping of similar characters to their stem, e.g., *ä, à, á, å, æ* to *a*. Furthermore, to cope with the fact that many artists use nicknames or abbreviations of their real names, an *approximate string matching* method was applied. According to [Cohen et al., 2003], the *Jaro-Winkler similarity* is well suited for personal names since it favors strings that match from the beginning for a fixed prefix length, e.g., "Edu Falaschi" vs. "Eduardo Falaschi", singer of the Brazilian band *Angra*. More precisely, a *level two distance function* based on the Jaro-Winkler distance metric was used. This means that the two strings to compare are broken into substrings (first, middle, and last names), and the string similarity is given by the combined similarities between each pair of tokens. It is assumed that two strings are equal if their Jaro-Winkler similarity is above 0.9. The Jaro-Winkler similarities were computed using the open source toolkit *SecondString* [sec, 2008].

Precision and Recall

Precision and recall of the predicted (member, instrument)-pairs on the ground truth was measured to evaluate the approach proposed in Subsection 3.3.4 for different query schemes. Such a (member, instrument)-pair is only considered correct if both the member and the instrument are predicted correctly. According to preliminary experiments, a filtering threshold of $t_{DF} = 0.25$ seemed to represent a good trade-off between precision and recall.

Given the set of correct (band member, instrument)-assignments T according to the ground truth and the set of assignments predicted by the approach under evaluation P, precision and recall are defined as $p = \frac{|T \cap P|}{|P|}$ and $r = \frac{|T \cap P|}{|T|}$, respectively. The results given in Table 5.14 are the average precision and recall values, over all bands in the ground truth set, using a filtering threshold of $t_{DF} = 0.25$. The first value indicates the precision, the second the recall.

Upper Limits for the Recall

Since the proposed approach relies on the availability of information on the retrieved Web pages, there exists an upper bound for the achievable performance. A band member that never occurs in the set of the 100 top-ranked Web pages of a band obviously cannot be detected by the approach. As knowing these upper bounds is crucial to estimate the goodness of the recall values presented in Table 5.14, the author analyzed how many of the actual band members given by the ground truth occur at least once in the retrieved Web pages, i.e., for every band b, the recall on the ground truth of the n-grams extracted

[3]For example, the drummer of the band *Tiamat*, *Lars Sköld*, is often referred to as *Lars Skold*.

Precision/Recall on *C51a240m*			
	exact	similar char	L2-JaroWinkler
M	46.94 / 32.21	50.27 / 34.46	53.24 / 35.95
MR	42.49 / 31.36	45.42 / 33.86	48.20 / 35.32
MM	43.25 / 36.27	44.85 / 37.23	47.44 / 37.55
LUM	32.48 / 27.87	33.46 / 29.06	34.12 / 29.06
Precision/Recall on *C51a499m*			
	exact	similar char	L2-JaroWinkler
M	63.16 / 23.33	68.16 / 25.25	72.12 / 26.38
MR	52.42 / 21.33	55.63 / 23.12	59.34 / 24.82
MM	60.81 / 26.21	63.66 / 27.45	67.32 / 27.64
LUM	43.90 / 19.22	44.88 / 19.75	46.80 / 20.08

Table 5.14: Overall precision and recall of the predicted (member, instrument)-pairs, in percent, for different query schemes and string distance functions on collections *C51a240m* and *C51a499m*.

Upper Limits for the Recall on *C51a240m*			
	exact	similar char	L2-JaroWinkler
M	56.00	57.64	63.44
MR	50.28	53.53	60.92
MM	58.12	59.69	66.33
LUM	55.80	58.62	66.26
Upper Limits for the Recall on *C51a499m*			
	exact	similar char	L2-JaroWinkler
M	52.97	55.15	62.01
MR	47.41	49.59	56.29
MM	56.40	57.62	64.08
LUM	55.21	57.27	64.11

Table 5.15: Upper limits for the recall achievable on collections *C51a240m* and *C51a499m*.

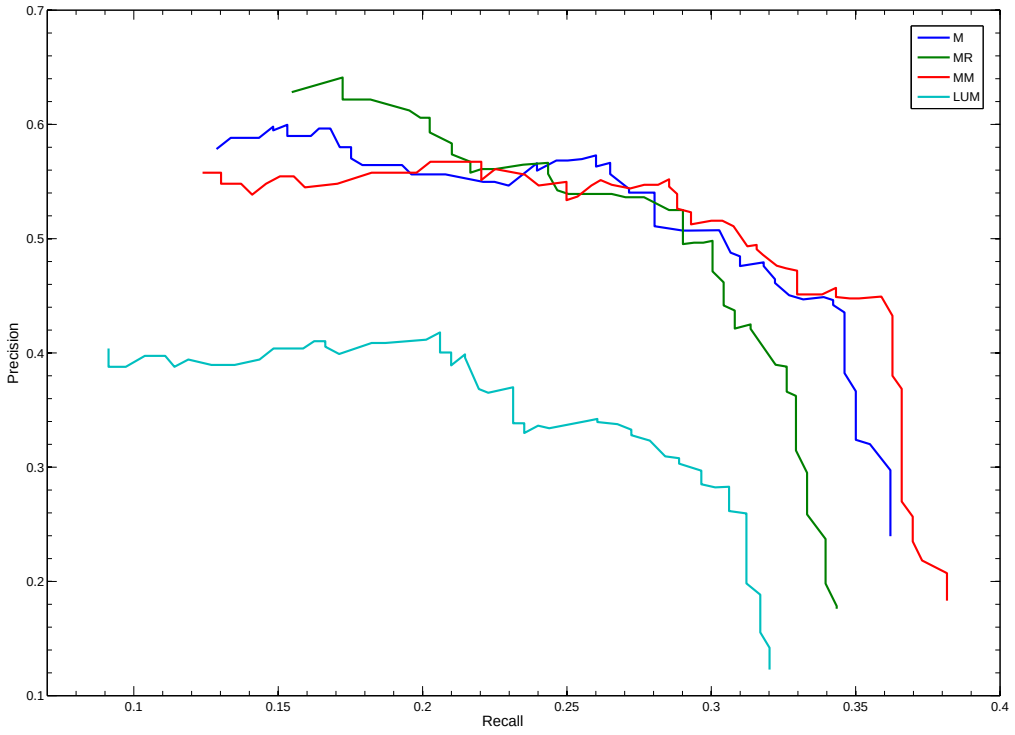


Figure 5.8: Precision/recall-plot for the band members and instrumentation detection approach on collection *C51a240m*, using exact string matching.

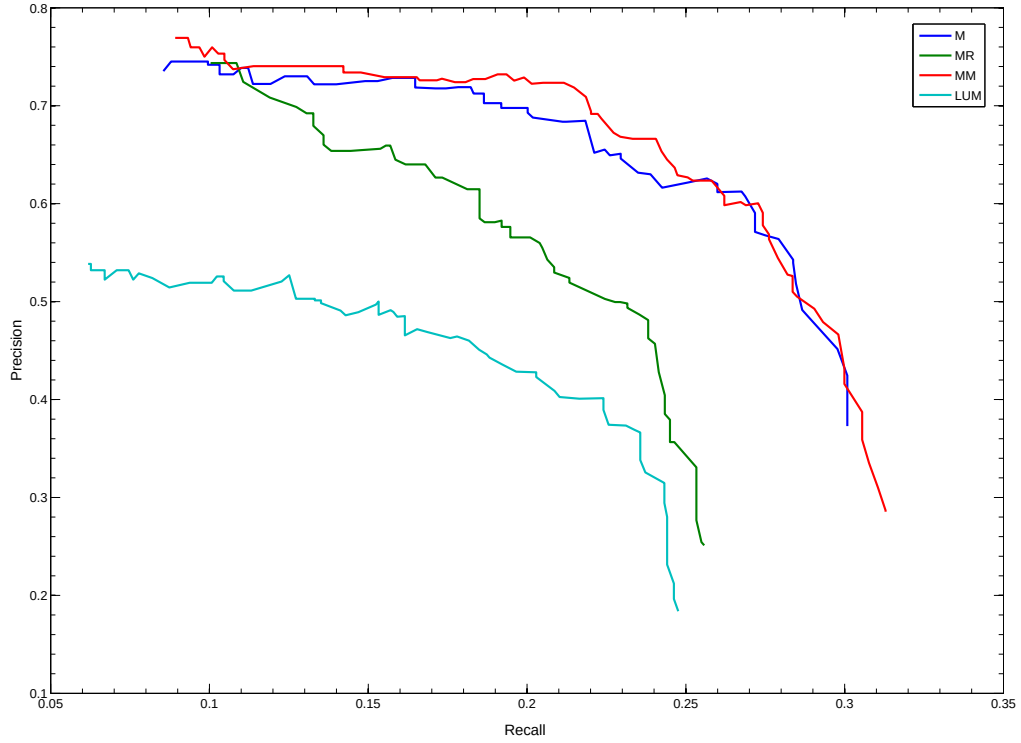


Figure 5.9: Precision/recall-plot for the band members and instrumentation detection approach on collection *C51a499m*, using exact string matching.

from b's Web pages was calculated. This calculation was based only on the n-grams, without taking information about instruments into account. Furthermore, it was verified that no band members were erroneously discarded in the n-gram selection phase, cf. Subsection 3.3.4. The results of these upper limit calculations using each query scheme and string matching function are depicted in Table 5.15 for both collections *C51a240m* and *C51a499m*. The values are given in percent.

Influence of the Filtering Threshold t_{DF}

The author also investigated the influence of the filtering threshold t_{DF} on precision and recall. For this purpose, a series of precision and recall calculations for successively increasing values of t_{DF} between 0.0 and 1.0 with an increment of 0.01 was performed. The resulting precision/recall-plots can be found in Figures 5.8 and 5.9 for the collections *C51a240m* and *C51a499m*, respectively. In these plots, only the results for exact string matching are presented for reasons of lucidity. Employing the other two, more tolerant, string distance functions just shifts the respective plots upwards. Since using low values for t_{DF} does not filter out many potential band members, the recall values tend to be high, but at the cost of lower precision. In contrast, high values of t_{DF} heavily prune the set of (member, instrument)-predictions and therefore generally yield lower recall and higher precision values.

Results and Discussion

Taking a closer look at the overall precision and recall values given in Table 5.14 reveals that, for both data sets, the query scheme M yields the highest precision values (up to more than 72% on the collection *C51a499m* using Jaro-Winkler string matching), whereas the more specific scheme MM is able to achieve a higher recall on the ground truth (a maximum recall of nearly 38% on the data set *C51a499m* using Jaro-Winkler string matching). The LUM scheme performs worst, independent of the used collection and string distance function. The MR scheme performs better than LUM, but worse than M and MM with respect to both precision and recall.

Comparing the precision and recall values obtained for the data set *C51a240m* with those obtained for *C51a499m* shows that for *C51a499m* the recall drops considerably. This is no surprise as *C51a499* contains more than double the number of band members as *C51a240m* and also lists members who spent a very short time with a band. For the same reasons, precision is higher for the collection *C51a499m*, since obviously the chance of correctly predicting a member is larger for a larger ground truth set, given the same corpus of Web pages.

Interestingly, comparing the upper limits for the recall for the two collections, cf. Table 5.15, reveals that extending the set of the current band members with those who already left the band does not strongly influence the achievable recall, even though the number of band members in the ground truth

set increases from 240 to 499 when adding the former members. This is a strong indication that the 100 top-ranked Web pages of every band, which are used in the retrieval process, contain information on the current as well as on the former band members to almost the same extent. Therefore, it seems reasonable to conclude that using more than 100 Web pages is unlikely to increase the quality of the (member, instrument)-predictions.

Regarding Figures 5.8 and 5.9, which depict the influence of the filtering threshold t_{DF} on the precision and recall values for the data sets *C51a240m* and *C51a499m*, respectively, reveals that, for the data set *C51a240m*, the query schemes M, MR, and MM do not strongly differ with respect to the achievable performance. For the collection *C51a499m*, in contrast, the results for the scheme MR are considerably worse than that for M and MM. It seems that album reviews, which are captured by the MR scheme, are more likely to mention the current band members than the former ones. This explanation is also supported by the fact that the highest precision values on the data set *C51a240m* are achieved with the MR scheme. Furthermore, the precision/recall-plots illustrate the worse performance of the LUM scheme, independently of the filtering threshold t_{DF}.

Taking a qualitative look at the results, good performance was achieved for those bands whose principal members spent a long time in the band and are still members, regardless of the popularity of the band. For example, all (member, instrument)-pairs were correctly identified for the very famous *Iron Maiden*, but also for the less known *Edguy* and *Pink Cream 69*. On the other hand, the approach obviously has problems with heavy band member fluctuations, especially if a very famous member left the band after years of participation. A good example of this is *Nightwish*, whose long-term singer *Tarja Turunen* left the band in 2006.

To summarize, taking the upper limits for the recall into account, cf. Table 5.15, the recall values achieved with the proposed approach, as given in Table 5.14, are quite promising. Overall, the query scheme M yields the highest precision, while the scheme MM yields the highest recall.

5.3.5 Album Covers

To get a first impression of the performance of image search engines offered by Web search providers when it comes to determining album cover artwork, preliminary experiments were conducted using collection *C255b*. These experiments are described in the following. Subsequently, the image retrieval approaches relying on distance measurement between <img> tags and textual identifiers on Web pages, as elaborated in Subsection 3.3.5, and the approach of predicting the image whose color histogram is most similar to the average histogram of all potential images, cf. Subsection 3.3.5, are investigated.

Image Search Engines

For the albums in collection *C255b*, the image search engines of A9.com and Google were queried using the schemes C and CA. Taking the top-ranked image returned by the search engine gives a baseline for evaluating the content-based filtering techniques described in Subsection 3.3.5. However, it is often quite difficult to figure out whether an album cover is correct or not, for example, due to different versions for different sales regions, covers that became censored after release, or remastered versions with new covers. For these reasons, automated evaluation is not feasible. Therefore, the author had to manually inspect each cover retrieved, which made evaluation a quite labor-intensive task.

Results and Discussion

Table 5.16 shows the evaluation results for the query schemes C and CA for the two search engines. The baseline approach simply regards the top-ranked image returned by the search engine as prediction. The upper part of the table illustrates the results for the scheme C, the lower part those for CA. All predicted cover images were manually classified into one of the following categories.

- correct cover image (column *correct*)
- image shows the album cover, but does not fit to the cover dimensions (column *dim.err.*)
- image shows a cover of another album or single by the same artist (column *other*)
- image shows a scanned compact disc (column *scanned*)
- image shows other artist-related material, e.g., a picture of the artist (column *related*)
- image is just completely wrong (column *wrong*)
- no image was found (column *not found*)

The values in Table 5.16 indicate the fraction of the items in each category on the total collection, in percent. *SCD* denotes the filtering of scanned compact discs, as described in Subsection 3.3.5. Results for applying filtering of non-quadratic images, cf. Subsection 3.3.5, can be found in the rows labeled *quad filtering*. It can be seen that Google's image search engine generally performs better than A9.com image search. Moreover, using CA instead of C does not only eliminate images of scanned compact discs, but unfortunately also decreases the number of found album cover images considerably. Therefore, the author concludes that improvements are unlikely to be achieved by constraining the search with additional keywords other than `cover`.

Moreover, Table 5.16 shows that rejecting all returned images with non-quadratic dimensions, within a

tolerance of 15%, yields an increase in accuracy by at least 3 percentage points for both query schemes and both search engines. Applying the circle detection technique to filter images of scanned compact discs further improves accuracy, especially for the C scheme in conjunction with Google. Using the quadratic dimension constraint together with the circle detection approach improves results from a baseline of 78% to 83% using Google and from 63% to 68% using A9.com.

		C						
		correct	dim.err.	other	scanned	related	wrong	not found
Google	1st (baseline)	77.78	0.44	6.22	2.22	2.22	2.22	8.89
	quad filtering	80.89	0.89	5.33	2.22	1.33	2.22	7.11
	quad + SCD	82.67	0.89	5.78	0.00	1.33	2.22	7.11
A9.com	1st (baseline)	63.11	4.89	4.89	1.33	5.78	5.33	14.67
	quad filtering	68.44	1.33	6.22	2.67	0.89	2.67	17.78
		CA						
		correct	dim.err.	other	scanned	related	wrong	not found
Google	1st (baseline)	63.11	1.33	5.78	0.44	1.78	4.44	23.11
	quad filtering	67.56	1.33	4.44	0.00	0.44	3.56	22.67
A9.com	1st (baseline)	56.44	1.78	7.11	0.44	2.22	3.56	28.44
	quad filtering	60.00	1.33	6.67	0.00	0.00	2.67	29.33

Table 5.16: Evaluation results for album cover detection approaches on collection *C225b*.

Image Retrieval Based on Text Distance and Content Analysis

The second set of experiments was conducted on the commercial collection *C3311b*. This collection comprises albums by various artists from all around the world. Thus, it should give more detailed insights into the behavior of the approaches to album cover retrieval on a broader spectrum of album cover artwork. As described in Subsection 3.3.5, a full inverted index including the plain text and the HTML tags of up to 100 top-ranked Web pages retrieved for queries to Google was created. Predicting one image as the album cover is then performed by selecting either the most average image with respect to color histogram similarity or the image whose reference in the Web page shows minimal character or tag distance to occurrences of artist and album names. Again, the author had to laboriously investigate each predicted album manually.

Results and Discussion

The results of these evaluation experiments can be found in Table 5.17. Non-quadratic image filtering was used for all approaches, except for the Google baseline, which always predicts the top-ranked image returned by the search engine. It can be seen that for the multifaceted collection *C3311b*, overall, only about 60% of the predicted album covers are correct. The main reason for this is the high amount of

album names (21%–26%) for which no corresponding cover image could be found. This suggests that, even in the best case, accuracies of more than about 75% seem unrealistic for real world collections. After all, for cover images appearing on Web pages indexed by Google, the experiments showed that accuracies can be improved by 3 percentage points using tag distance for image selection and filtering of non-quadratic images and scanned compact discs.

		C						
		correct	dim.err.	other	scanned	related	wrong	not found
Google	1st (baseline)	56.69	1.48	7.76	0.15	1.63	5.80	26.46
Index	avg histogram (without SCD)	9.50	0.00	17.01	0.41	3.73	62.66	6.22
	avg histogram (with SCD)	9.96	0.00	17.43	0.41	3.32	61.83	6.64
	tag distance (without SCD)	55.19	0.00	4.56	0.41	2.49	11.62	25.31
	tag distance (with SCD)	58.88	0.00	5.62	0.18	1.99	13.59	19.57
	char distance (without SCD)	56.58	0.00	9.24	0.28	1.96	10.92	20.73
	char distance (with SCD)	57.87	0.00	10.96	0.28	3.09	12.36	15.17

Table 5.17: Evaluation results for album cover detection approaches on collection *C3311b*.

5.3.6 Visualizing Artist-Related Web Pages with the COB

To investigate the usefulness of the COB, cf. Subsection 4.2.6, for gaining a quick overview of a set of Web pages and efficiently browsing within this set, a small user study was conducted by the author of this PhD thesis. While the assessment reported in Subection 5.3.3 focused on evaluating the quality of the descriptive terms for different term weighting functions, the study elaborted here primarily addresses ergonomic aspects of the user interface. To this end, the author formulated the following tasks, which he believes to be important for these purposes, and evaluated them in a quantitative manner.

1. Which are the five top-ranked terms that occur on the Web pages mentioning "Iron Maiden"?
2. Indicate the number of Web pages containing all of the terms "Iron Maiden", "metal", and "guitar".
3. Show a list of Web pages that contain the terms "Iron Maiden" and "british".
4. Considering the complete set of Web pages, which are the three terms that co-occur on the highest number of Web pages?
5. How many Web pages contain the terms "Iron Maiden" and "metal", but not the term "guitar"?
6. Display a list of audio files available at Web pages containing the term "Iron Maiden".

7. Which terms co-occur on the set of Web pages that contains the highest number of image files in hierarchy level three?
8. Indicate the URL of one particular Web page that contains image files, but no video files.
9. How many Web pages does the complete collection contain?
10. Find one of the deepest elements in the hierarchy and select it.
11. Generate a new visualization using only the Web pages on which the terms "bass" and "heavy metal" co-occur.

Tasks 1–8 are general tasks that are likely to arise when analyzing and browsing collections of Web pages. In particular, Tasks 1–5 address the co-occurring terms, whereas Tasks 6–8 deal with the multimedia content extracted from the Web pages. In contrast, Tasks 9–11 relate to the structure of the Sunburst tree.

After having explained the interaction functionalities provided by the COB to the participants, they had five minutes to explore the user interface themselves with a visualization for the artist *Britney Spears*. During this warm-up, the participants were allowed to ask questions. After the exploration phase, they were presented the visualization obtained for the Web page collection of the band *Iron Maiden*, cf. Figure 4.10. The constraints for creating the visualization, cf. Subsection 4.2.5, had been set to the following values: $max\_subnodes = 8$, $max\_depth = 8$, $min\_agl\_ext = 3.0$. The participants were then consecutively asked each of the questions, and the time they needed to finish each task was measured. Each participant had a maximum of three minutes to complete each task.

The number of participants in the user study was six (five males, one female). All of them were computer science or business students at the *Johannes Kepler University Linz* and all stated to have a moderate or good knowledge of user interfaces and to be very interested in music. All participants performed the user study individually, one after another. The study was carried out on a *Pentium 4* 3GHz with 2GB RAM and an *nVidia GeForce 6600 GT* graphics card running under *ubuntu Linux*.

Results and Discussion

Table 5.18 shows the times needed by the participants to finish each task. Inverse numbers indicate that the given answer was wrong. Furthermore, the average time required to give the **correct** answer is indicated for each task. In general, the tasks related to structural questions were answered in a shorter time than those related to browsing the collection. Among the structural questions, solely Task 11 required a quite high average time. This can be explained by the fact that the term "bass" was not easy to find on all layers. The same holds for the term "british", requested in Task 3.

Task	1	2	3	4	5	6	7	8	9	10	11
Participant A	28	13	45	47	36	61	172	*180*	2	12	*25*
Participant B	69	23	46	52	14	15	68	76	*6*	12	62
Participant C	15	3	39	27	22	3	34	68	1	9	31
Participant D	132	1	57	30	117	14	43	*180*	5	12	40
Participant E	110	9	16	8	*163*	7	12	148	2	38	74
Participant F	36	14	21	46	44	12	79	*180*	3	5	61
Mean	**65**	**11**	**37**	**35**	**47**	**19**	**68**	**97**	**3**	**15**	**54**

Table 5.18: For each participant, the time needed to finish each task of the COB user study, measured in seconds.

For the questions related to browsing in the hierarchy, it was observed that tasks requiring intensive rotation of the Sunburst stack (1, 3, 4, 5, 7) yielded worse results than those for which this was not necessary (2, 6). In general, users spent a lot of time rotating the Sunburst stack to a position at which the label of the selected arc was readable. This process should be automatized in future versions of the COB.

The relatively high average time required to perform the first task may be attributed to the fact that most participants needed some seconds to get used to the new visualization of *Iron Maiden* after having explored the Web pages of *Britney Spears* in the exploration phase. Task 2 was successfully finished quite fast (in 11 seconds on average) by all participants. This may be explained by the fact that the combination of the terms "Iron Maiden", "metal", and "guitar" was one of only two term combinations that formed a third hierarchy level in the visualization. In spite of the fact that Task 3 was solved in only 37 seconds on average, four participants (A, B, C, D) had problems locating the arc "british" since it was hardly perceivable due to its position behind a much higher arc (on all of the three layers). Interestingly, the average time needed for Task 4 was much higher than the average time needed for Task 2, despite the fact that both required finding the same arc. As for Task 5, the participants A, B, C, and F solved it in a reasonable time, whereas participants D and E were not sure which number to subtract from which other. Task 6 posed no difficulties to most participants. Only participant A chose the time-consuming solution of first generating a new visualization for the Web pages containing the term "Iron Maiden" and then pressing the key *A*. Solving Task 7 took the second highest average time since it required finding and navigating to the Sunburst illustrating the amount of image files and comparing the heights of all arcs in hierarchy level three of this Sunburst. As for Task 8, it yielded the worst results as three of the participants were not able to solve this task correctly. The main problem was that no arc on the video layer had a height of zero, which confused most participants. It was obviously not clear to them that a positive height of an arc on the video layer does not necessarily indicate that each Web page represented by this arc offers video content. After all, two participants solved this task in less than one and a half minutes.

To conclude, the user study showed that the COB can be efficiently used for tasks related to browsing sets of Web pages. Although barely comparable to the user study in [Kobsa, 2004] on similar tree visualization systems due to a different application scenario, a very rough comparison of the average required time over all tasks shows that this time is much shorter for the COB (45 seconds) than for the best performing system of [Kobsa, 2004] (101 seconds). Therefore, the results of the conducted study are promising. However, it can be stated that the user interaction functionalities provided by the COB need some improvements.

Chapter 6

AGMIS: An Automatically Generated Music Information System Based on Information from the Web

In this chapter, details about the implementation of the *Automatically Generated Music Information System* (AGMIS) are presented. Since AGMIS serves as the core application to demonstrate the techniques elaborated for this PhD thesis, it integrates functionalities for Web page retrieval and indexing, information extraction, data storage, and data access via a Web user interface. A schematic illustration of the complete data processing steps performed by AGMIS is given in Figure 6.1. Each of the corresponding functional units will be presented in the following. First, however, some details a bout the used artist collection that served as input are given.

6.1 Artist Collection

The first task in the data processing pipeline, cf. Figure 6.1, is to gather a sufficiently large list of artist names, for which information will be offered by the AGMIS. To this end, the author extracted from *AMG* [amg, 2007a] more than 630,000 music artists and bands, organized in 18 different genres. In a subsequent data preprocessing step, all artists within each genre mapping to identical strings after *non-character removal*[1] were discarded once from the set. Table 6.1 lists the genre distribution of the remaining 636,475 artists according to AMG, measured as absolute number of artists in each genre and as percentage in the complete collection. The notably high amount of artists in the genre "Rock" can be explained by the large diversity of different music styles within this genre. In fact, taking a closer look at the artists subsumed in the genre "Rock" reveals pop artists as well as death metal bands. Nevertheless, gathering artist names from AMG seemed the most reasonable solution to obtain a large corpus.

The sole input to the following data acquisition steps is the list of extracted artist names, except for the prototypicality estimation, which also requires genre information, and for the album cover artwork, which requires album names.

[1] This filtering was performed to cope with different spellings for the same artist, e.g., "B.B. King" vs. "BB King".

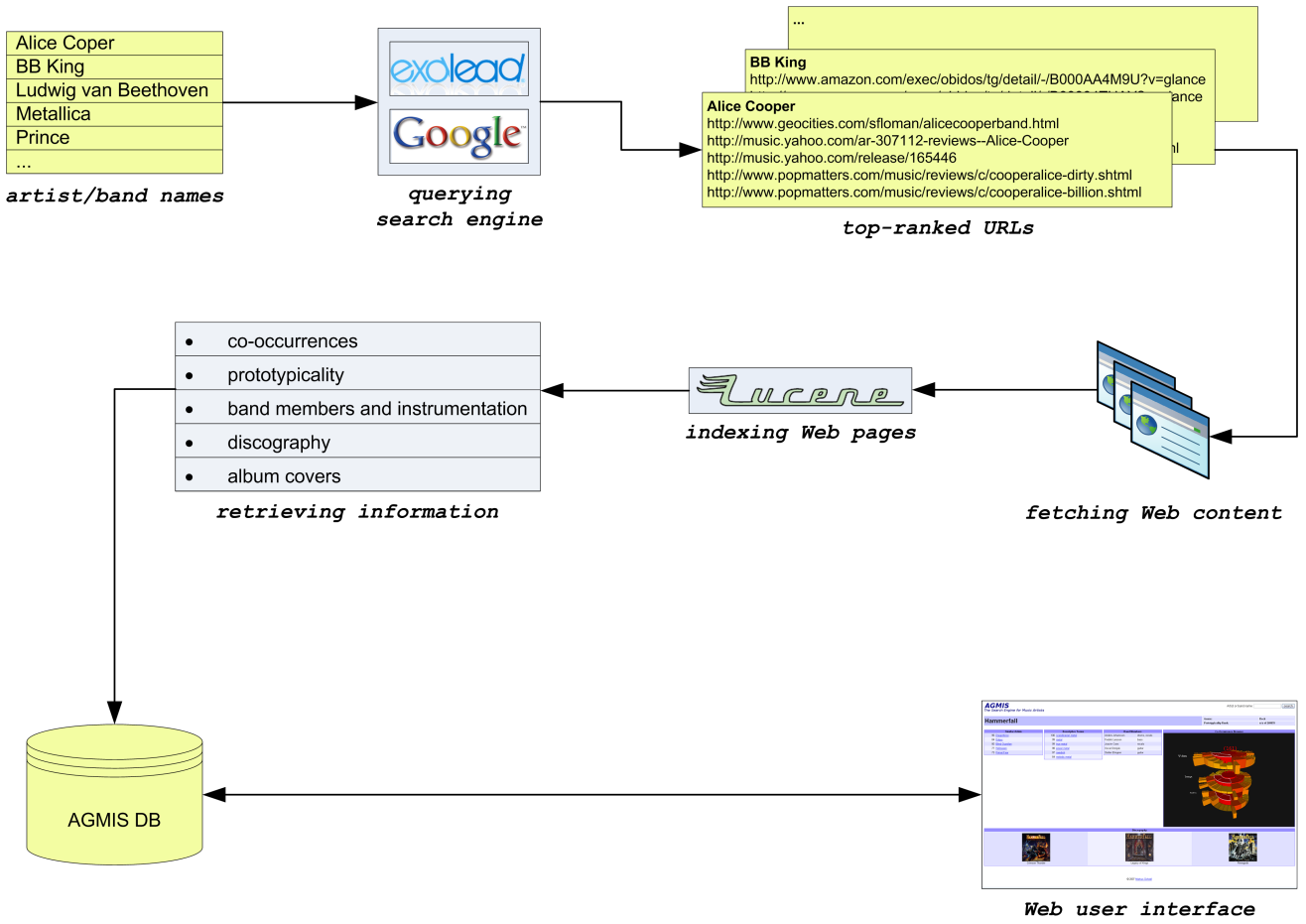


Figure 6.1: Data processing diagram of AGMIS.

6.2 Data Acquisition

The data acquisition process can be broadly divided into the three phases *querying* the search engine for the URLs of artist-related Web pages, *fetching* the HTML documents available at the retrieved URLs, and *indexing* the contents of these documents. Each of these phases is detailed in the following.

6.2.1 Querying

The author started querying the *exalead* [exa, 2007] search engine for URLs of up to 100 top-ranked Web pages for every artist in the collection using the query scheme `"artist name" NEAR music` on the 5th August 2007. Even though it can be assumed that Google would give better results, Google's policy of restricting automated queries to 1,000 per day made its use infeasible. Despite the fact that for some tasks, query schemes other than M performed best, cf. Section 5.3, M was chosen since it is the least restrictive scheme, which is important as the huge collection contains a large amount of largely unknown artists. The querying process took approximately one month. Its outcome was a list of 26,044,024 URLs, which had to be fetched.

6.2.2 Fetching

To fetch the huge amount of more than 26 millions of Web pages, the author of this PhD thesis implemented a fetcher incorporating a load balancing algorithm to avoid excessive bandwidth consumption of servers frequently occurring in the URL list. The fetching process took approximately four and a half months and was performed in the time between September 2007 and January 2008. It yielded a total of 732.6 gigabytes of Web pages. The Web pages from one specific host were set on a black list, thus not fetched, since the respective Web master strongly discouraged the author from automatically retrieving content provided by this host. That is why the total number of retrieved Web pages, given in Table 6.3, differs from the 26,044,024 returned for the exalead queries.

Some statistics concerning the retrieved Web pages can be found in Tables 6.2, 6.3, and 6.4. Table 6.2 shows, for each genre, the amount of artists for which not a single Web page could be determined by the exalead search engine, i.e., artists with a page count of zero. These amounts are given as absolute numbers and as relative percentages. Not very surprisingly, the percentage is highest for the genres "Latin" and "World" (nearly 30% of 0-PC-artists), which comprise many artists known only in regions of the world that are lacking a broad availability of Internet access. In contrast, a lot of information seems to be available online for artists in the genres "Electronica" and "Rap" (about 10% of 0-PC-artists). Table 6.3 depicts, for each genre, the number of Web pages retrieved and the amount of retrieved pages with a length of zero, i.e., pages that were empty or could not be fetched for some reason. These amounts are again given as absolute numbers and as percentages. Since the main reason for the occurrence of such pages were server errors, e.g., server time outs, their relative frequencies are largely genre-independent, as it can be seen in the last column of Table 6.3. Table 6.4 shows the median and arithmetic mean of the page counts returned by exalead for the artists in each genre. From this table, it can be seen again that artists in the genres "Latin", "Jazz", and "Gospel" tend to be underrepresented on the Web. Note that the median page counts according to exalead (column *Median PC*) correspond to the median of the number of retrieved Web pages since they are all below 100, which is the maximum number of Web pages retrieved. The last column gives the arithmetic mean of the number of retrieved Web pages for each genre.

6.2.3 Indexing

To create a *full inverted document index* of the retrieved Web pages, i.e., an index that not only stores a mapping `term` $\mapsto$ `document*`, but a mapping `term` $\mapsto$ `(document,position+)*`, the open source indexer *Lucene Java* [luc, 2008] was taken as a basis and adapted to suit the HTML format of the input documents and the requirements for efficiently extracting the desired artist-related pieces of information.

Genre	Artists	Percentage
Avantgarde	4,469	0.70 %
Blues	13,592	2.14 %
Celtic	3,861	0.61 %
Classical	11,285	1.77 %
Country	16,307	2.56 %
Easy Listening	4,987	0.78 %
Electronica	35,250	5.54 %
Folk	13,757	2.16 %
Gospel	26,436	4.15 %
Jazz	63,621	10.00 %
Latin	33,797	5.31 %
New Age	13,347	2.10 %
Rap	26,339	4.14 %
Reggae	8,552	1.34 %
RnB	21,570	3.39 %
Rock	267,845	42.08 %
Vocal	11,689	1.84 %
World	59,771	9.39 %
Total	**636,475**	**100.00 %**

Table 6.1: List of genres used in AGMIS with the corresponding number of artists and their share in the complete collection, in percent.

Genre	Artists	0-PC-Artists	0-PC-Artists (%)
Avantgarde	4,469	583	13.05 %
Blues	13,592	2,003	14.74 %
Celtic	3,861	464	12.02 %
Classical	11,285	1,895	16.79 %
Country	16,307	2,082	12.77 %
Easy Listening	4,987	865	17.35 %
Electronica	35,250	3,101	8.80 %
Folk	13,757	2,071	15.05 %
Gospel	26,436	5,597	21.17 %
Jazz	63,621	10,866	17.08 %
Latin	33,797	9,512	28.14 %
New Age	13,347	2,390	17.91 %
Rap	26,339	2,773	10.53 %
Reggae	8,552	1,320	15.43 %
RnB	21,570	2,817	13.06 %
Rock	267,845	39,431	14.72 %
Vocal	11,689	1,988	17.01 %
World	59,771	17,513	29.30 %
Total	**636,475**	**107,271**	**16.85 %**

Table 6.2: Amount of artists for which no Web pages were found (Zero-Page-Count-Artists).

Genre	Artists	Retrieved Pages	0-Length-Pages	0-Length-Pages (%)
Avantgarde	4,469	204,870	32,704	15.96 %
Blues	13,592	554,084	89,832	16.21 %
Celtic	3,861	136,244	23,627	17.34 %
Classical	11,285	509,269	99,181	19.48 %
Country	16,307	696,791	116,299	16.69 %
Easy Listening	4,987	187,749	32,758	17.45 %
Electronica	35,250	1,973,601	317,863	16.11 %
Folk	13,757	544,687	89,385	16.41 %
Gospel	26,436	876,017	142,690	16.29 %
Jazz	63,621	2,306,785	361,160	15.66 %
Latin	33,797	866,492	139,660	16.12 %
New Age	13,347	488,799	82,075	16.79 %
Rap	26,339	1,322,187	223,052	16.87 %
Reggae	8,552	377,355	58,180	15.42 %
RnB	21,570	898,787	141,339	15.73 %
Rock	267,845	12,058,028	1,908,904	15.83 %
Vocal	11,689	461,374	77,073	16.71 %
World	59,771	1,577,769	257,649	16.33 %
Total	**636,475**	**26,040,888**	**4,193,431**	**16.10 %**

Table 6.3: Retrieved Web pages and empty Web pages.

Genre	Median PC	Mean PC	Mean # Pages Retrieved
Avantgarde	29	14,969	46
Blues	18	2,893	40
Celtic	25	5,415	35
Classical	27	4,149	45
Country	22	2,562	42
Easy Listening	14	4,808	37
Electronica	65	31,366	56
Folk	18	5,166	39
Gospel	8	4,791	33
Jazz	13	6,720	36
Latin	4	19,384	25
New Age	13	12,343	36
Rap	37	38,002	50
Reggae	22	16,000	44
RnB	17	17,361	41
Rock	21	16,085	43
Vocal	15	10,421	39
World	4	14,753	26
Total	**16**	**15,120**	**40**

Table 6.4: Median and mean of available Web pages (according to page counts) and mean of actually retrieved Web pages.

Although indexing seems to be a straightforward task at first glance, this is not the case, especially not if the input consists of HTML documents. During indexing, some heavily erroneous HTML files were encountered, which caused Lucene to hang or crash, and thus required special handling. More precisely, some HTML documents showed a size of tens of megabytes, but were largely filled with escape characters or numerously repeated sequences of error messages. To resolve these problems, a size limit of 5 megabytes for the HTML files to index was introduced. Additionally, a 255-byte-limit for the length of each token was used.

In total, three indexes have been created. For the first one, each term occurring on any Web page was indexed. Thus, neither stopping, nor stemming, nor casefolding was applied. This first index was mainly used for band member detection, cf. Subsection 3.3.4.The size of the optimized, compressed index is 228 gigabytes. A second index using only the terms in the music dictionary has been created to generate term profiles for the purpose of artist tagging, i.e., to find descriptive terms for an artist, and to calculate artist similarities. The size of this index is 28 gigabytes. The third index was created using all terms except stop words, applying Snowball stemming, and performing casefolding. Its size amounts to 214 gigabytes.

6.3 Database Design

The backend of the AGMIS system constitutes a relational *MySQL* [mys, 2008] database. A Web service implemented by the author using *Java servlet* technology then extracts the information requested by the user via the Web user interface from the AGMIS database. Figure 6.2 depicts the entity relationship diagram of the database.

6.4 Information Extraction

In the following, some remarks on the approaches used to extract the various pieces of information for the AGMIS artist collection are given.

As for estimating artist similarity, TF·IDF vectors of the terms given by the music dictionary, cf. Subsection 3.2, were first calculated for each artist in the collection. In the next step, an artist similarity matrix was generated. Computing the complete 636,475×636,475 similarity matrix requires 202,549,894,575 pairwise similarity calculations between TF·IDF vectors. Although performing this number of calculations is feasible in reasonable time on a current personal computer in regard to computational power, the challenge is to have the required term vectors in memory when they are needed. As the size of the complete similarity matrix amounts to nearly 800 gigabytes, even when storing symmetric elements only once, it is not possible to hold all data in memory. Therefore, the author first split the 636,475×636,475

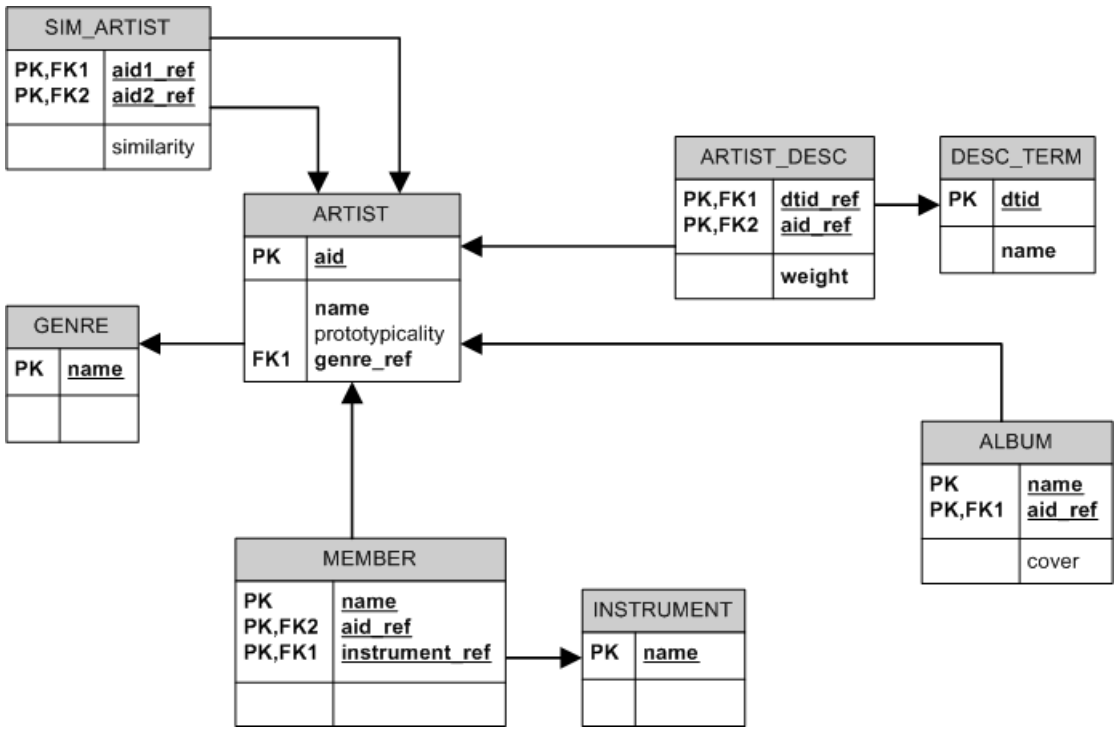


Figure 6.2: Entity relationship diagram of the AGMIS database.

matrix into 50 rows and 50 columns, yielding 1,275 submatrices when storing symmetric elements only once. Each such submatrix requires 622 megabytes and thus fits well into memory. Artist similarities were then calculated between the 12,730 artists in each submatrix, processing one submatrix at a time. Aggregating these submatrices, individual artist similarity vectors were extracted, and the most similar artists for each artist in the collection were selected and inserted into the database.

Due to the fact that using Lucene to process the amount of queries necessary for the generation of a complete artist co-occurrence matrix would have taken too much time, the author opted for a combination of *page counts-based ranking*, cf. Subsection 3.3.1, and *penalization of exorbitant popularity*, cf. Subsection 3.3.2, to estimate artist prototypicality. To this end, exalead page counts were determined for queries of the following form:

- `"artist name" AND music`
- `"artist name" AND "genre name" AND music`

The former scheme gives an estimate of the overall, genre-independent popularity of the artist under consideration, whereas the latter estimates the importance of the artist for a specific genre. These two kinds of page counts thus correspond to *genre prototypicality* and *overall prototypicality* according to Subsection 3.3.2. Eventually, the final ranking of an artist a with respect to a genre g is obtained via Equation 6.1, where pc_a is the page count obtained for the first query scheme, and $pc_{a,g}$ is the page

count obtained for the second query scheme.

$$r(a,g) = pc_{a,g} \cdot \left(\log \frac{1}{pc_a + 1} \right)^2 \tag{6.1}$$

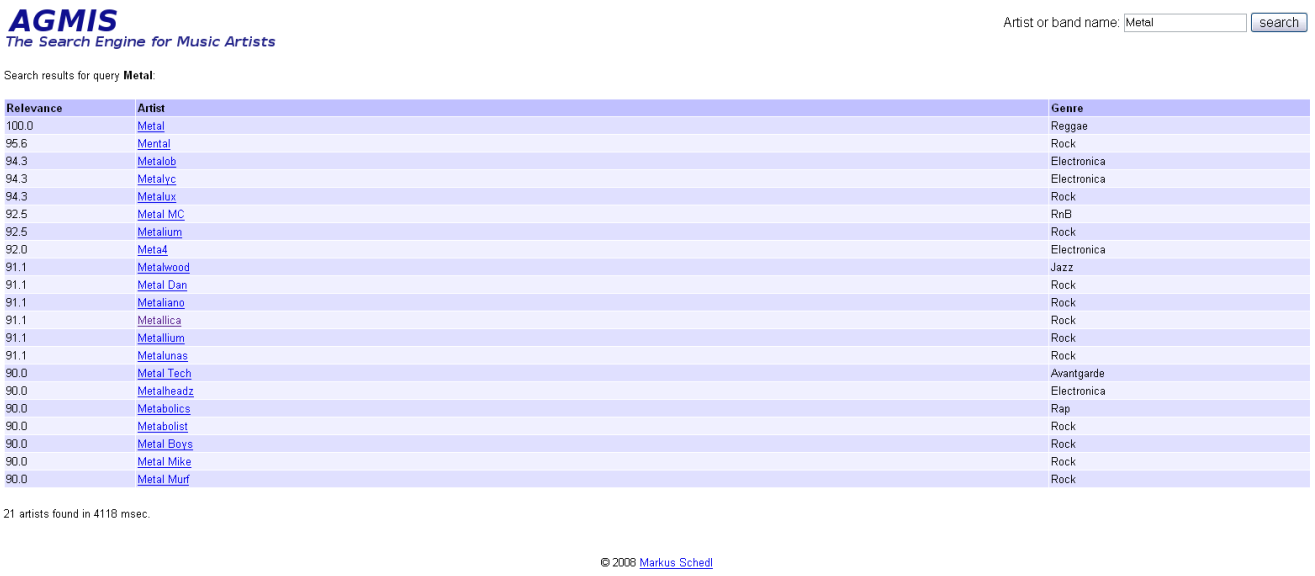
Band member and instrumentation detection was also performed using query scheme M, which seemed reasonable since no significant differences between the M and MM schemes could be made out in the corresponding evaluation experiments, cf. Subsection 5.3.4. Since the named entity detection and the rule-based linguistic analysis steps in band member detection require information on term capitalization and on exact term positions, a full inverted index without stopping, stemming, and casefolding was used. The respective pieces of information were then extracted from this index via the approach presented in Subsection 3.3.4.

Retrieving album cover artwork was done using the full inverted index of the plain text and the HTML tags. On the set of potential album cover images given by all image links occurring in the retrieved Web pages of the artist under consideration, filtering of non-quadratic images and scanned compact discs was performed to obtain a preselection. Subsequently, the image with the minimal sum of the tag distances between its <img> tag and the occurrences of artist and album name was selected, as described in Subsection 3.3.5. For artists with a page count of zero, the baseline approach of querying Google's image search engine was used as fallback.

6.5 Web User Interface

The pieces of information extracted from the artist-related Web pages and inserted into the AGMIS database are offered to the user of the system via a Web service build on *Java servlet* and *Java applet* technology. The home page of the AGMIS Web site reflects a quite simple design, like the one used by Google. Besides a brief explanation of the system, it only displays a search form, where the user can enter an artist or band name. To allow for fuzzy search, the string entered by the user is compared to the respective database entries using *Jaro-Winkler similarity*, cf. [Cohen et al., 2003]. The user is then provided a list of approximately matching artist or band names, from which he/she can select one. An exemplary search for "Metal" yields the list of suggestions shown in Figure 6.3.

After the user has selected the desired artist, AGMIS delivers an artist information page. Figure 6.4 shows the upper part of such a page for the artist *B.B. King*. For reasons of lucidity, most of the discography and album cover information is excluded in this screenshot. On the top of the page, artist name, genre, and prototypicality rank are shown. Below this header, lists of similar artists, of descriptive terms, and of band members and instrumentation, where available and applicable, are shown. As a



AGMIS
The Search Engine for Music Artists

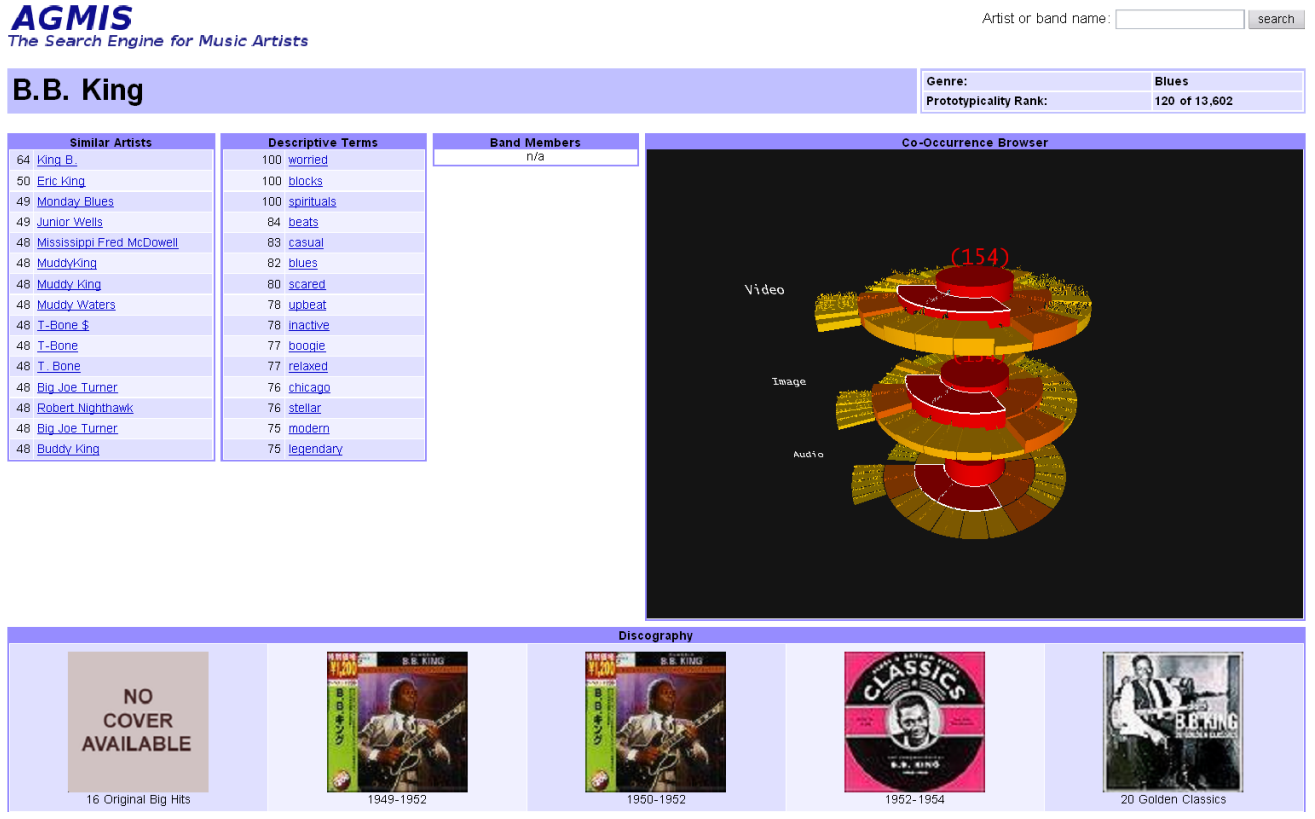
Artist or band name: Metal search

Search results for query **Metal**:

Relevance	Artist	Genre
100.0	Metal	Reggae
95.6	Mental	Rock
94.3	Metalob	Electronica
94.3	Metalyc	Electronica
94.3	Metalux	Rock
92.5	Metal MC	RnB
92.5	Metalium	Rock
92.0	Meta4	Electronica
91.1	Metalwood	Jazz
91.1	Metal Dan	Rock
91.1	Metaliano	Rock
91.1	Metallica	Rock
91.1	Metallium	Rock
91.1	Metalunas	Rock
90.0	Metal Tech	Avantgarde
90.0	Metalheadz	Electronica
90.0	Metabolics	Rap
90.0	Metabolist	Rock
90.0	Metal Boys	Rock
90.0	Metal Mike	Rock
90.0	Metal Murf	Rock

21 artists found in 4118 msec.

Figure 6.3: List of artists provided by AGMIS for the search term "Metal".



AGMIS
The Search Engine for Music Artists

Artist or band name: search

B.B. King

Genre:	Blues
Prototypicality Rank:	120 of 13,602

Similar Artists	
64	King B.
50	Eric King
49	Monday Blues
49	Junior Wells
48	Mississippi Fred McDowell
48	MuddyKing
48	Muddy King
48	Muddy Waters
48	T-Bone $
48	T-Bone
48	T. Bone
48	Big Joe Turner
48	Robert Nighthawk
48	Big Joe Turner
48	Buddy King

Descriptive Terms	
100	worried
100	blocks
100	spirituals
84	beats
83	casual
82	blues
80	scared
78	upbeat
78	inactive
77	boogie
77	relaxed
76	chicago
76	stellar
75	modern
75	legendary

Band Members
n/a

Co-Occurrence Browser

Discography

NO COVER AVAILABLE
16 Original Big Hits

1949-1952

1950-1952

1952-1954

20 Golden Classics

Figure 6.4: Part of the user interface provided by AGMIS for the artist *B.B. King*.

Figure 6.5: Complete artist information page returned by AGMIS for *B.B. King*.

matter of course, the information pages of similar artists are made available via hyperlinks. Moreover, it is also possible to search for artists via descriptive terms. By clicking on the desired term, AGMIS starts searching for artists that have this term within their set of highest ranked terms and subsequently displays a selection list, similar to the one shown in Figure 6.3. To the right of the lists described so far, the Co-Occurrence Browser is integrated into the user interface as a Java applet. Since the COB offers a 3D user interface, a fast graphics card is required to joyfully use it. The lower part of the artist information page is dedicated to discography information, i.e., a list of album names and album cover images are shown. Figure 6.5 depicts the complete artist information page returned by AGMIS for the artist *B.B. King*. In this screenshot, all 222 albums listed by AMG for *B.B. King* are illustrated.

6.6 Computational Complexity

Most tasks necessary for the creation of AGMIS were quite time-consuming. In the following, a quick overview of some running times is given to conclude this chapter.

The indexing process, the creation of artist term profiles, the calculation of term weights, and all information extraction tasks were performed on two standard personal computers with a Pentium 4 processor clocked at 3 GHz, 2 GB RAM, and a RAID-5 storage array totaling to 2 TB of usable space. In addition, a considerable amount of external hard disks serving as temporary storage facilities were required.

As already mentioned in the respective subsections, the process of querying the exalead search engine took about one month, and the load balanced fetching of the roughly 26 million URLs returned took about four and a half months. In Table 6.5, precise running times for indexing, information extraction, and database operation tasks are shown for those tasks for which the author measured the time.

Task	Running Time (secs)
Creating Lucene index using all terms (no stopping, no stemming, no casefolding)	218,681
Creating Lucene index using the music dictionary	211,354
Creating Lucene index using all terms (stopping, Snowball stemming, casefolding)	610,792
Computing the term weights (TF, DF, and TF·IDF)	514,157
Sorting the terms for each artist and each weighting function	13,503
Computing the artist similarity matrix via submatrices	2,489,576
Extracting artist similarity vectors from the submatrices	3,011,719
Estimating artist prototypicalities by querying exalead	4,177,822
Retrieving album cover artwork	6,654,703
Retrieving information on multimedia content for the COB	2,627,369
Retrieving band members and instrumentation for artists in genre "Rock"	213,570
Importing the 20 most similar artists for each artist into the AGMIS DB	356,195
Importing the 20 top-ranked terms for each artist into the AGMIS DB	3,649
Importing album names and covers into the AGMIS database	6,686

Table 6.5: Some running times of tasks performed for the creation of AGMIS.

Chapter 7

Conclusions

In this PhD thesis, approaches to automated extraction of information related to music artists from Web pages have been elaborated. The presented approaches address the problems of determining *similarities between music artists*, estimating the *prototypicality of an artist for a genre*, extracting *descriptive terms for an artist*, finding *band members and instrumentation of a music band*, and mining *images of album cover artwork*. After having described the corresponding information categories in Chapter 2, the author presented the information retrieval approaches elaborated in the context of this thesis in Chapter 3. These approaches were used to extract most of the pieces of information described in Chapter 2. In particular, the author proposed novel techniques to artist similarity measurement based on co-occurrences of artist names on artist-related Web pages, to prototypicality estimation based on backlink/forward link analysis with penalization of exorbitant artist popularity, to detect band members and their instruments based on natural language processing, and to retrieve images of album cover artwork based on content-based filtering and on image selection via distances between identifiers in the Web pages. In Chapter 4, information visualization methods to illustrate the various categories of artist-related information were presented. In particular, the *Continuous Similarity Ring*, the *Circled Fans*, and the *Stacked Three-Dimensional Sunbursts*, which have been developed by the author, were introduced. Chapter 5 reports on the various evaluation experiments conducted to investigate the approaches presented in the previous two chapters. Different collections and query schemes were used to this end. As one of the aims of this thesis was to integrate the extracted pieces of information into a music information system, Chapter 6 elaborated on the creation of AGMIS, the *Automatically Generated Music Information System*. AGMIS offers information on 636,475 music artists. In addition to the pieces of information extracted by the techniques presented in Chapter 3, AGMIS provides a variant of the *Co-Occurrence Browser*, which is a user interface to browse the artist-related Web pages retrieved.

To summarize the results of the conducted experiments, it was shown that co-occurrence analysis based on retrieved Web page content outperforms page counts-based co-occurrence analysis for the task of artist similarity estimation, not only in regard to scalability, but also in regard to accuracy. Furthermore, it could be shown that using specific combinations of query schemes for co-occurrence analysis performs better than the standard TF·IDF approach, provided that enough artist-related Web pages

are available.

The best results for artist prototypicality estimation were achieved using the approach based on back-link/forward link analysis and penalization of exorbitant artist popularity. In particular, genres containing many artist names that equal common speech words benefit from the penalization.

For the task of determining descriptive artist properties, the conducted user study showed that using document frequencies for term selection performed best. However, some of the terms attributed to the artists in the test collection were either too general ones or part of artist or album names and thus were not being considered very descriptive.

The approach to band member and instrumentation detection yielded remarkable results for the standard line-up of most rock bands, regardless of the popularity of the band under consideration.

As for determining images of album covers, it could be shown that filtering of non-quadratic images and images of scanned compact discs significantly raises the number of correctly predicted album cover images. Among the evaluated approaches to selecting the most probable cover image, predicting the image whose <img> tag shows the shortest distance to artist and album names in the HTML document yielded the best results.

Chapter 8

Outlook and Future Work

The techniques underlying the current implementation of AGMIS, which were elaborated in the context of this PhD thesis, although giving respectable results for the different information categories, still leave room for improvement in various directions.

As for the task of Web page retrieval, pursuing the strategy of focused crawling to harvest music-related Web pages would probably be superior to the approach of issuing queries to Web search engines. Thus, building a music-specific focused crawler is one of the next steps that should be taken. Doing so would presumably yield more accurate results, while at the same time limit Web traffic.

The information extraction techniques addressing the different information categories could be enhanced and refined as follows. In general, deep natural language processing techniques and more sophisticated approaches to named entity detection and machine learning could be employed to derive more specific information, especially in band member and instrumentation detection as well as to obtain detailed discography information. For example, extracting the release dates of albums would enable sorting the respective album covers according to year of release. Moreover, in the context of band member detection, temporal information would allow for creating complete band and artist histories as well as time-dependent relationship networks. Under the assumption that bands which share or shared some members are similar to some extent, such networks could even be used to derive a similarity measure. Taking these ideas a step further, automatically generated biographies would be the ultimate aim.

A further research direction in the context of album covers is the question whether artists that produce a certain style of music tend to favor certain styles in their album cover artwork. One could assume, for example, that artists producing dark or black metal tend to favor dark colors in their cover design. Automatically uncovering such relations, if they exist, would be at least entertaining.

Another interesting question is how to update the information in the system. In particular, automatically detecting new artists and bands as well as new album and song releases would certainly be a nice feature and should be feasible with sophisticated natural language processing techniques, which are tailored to the needs of music information retrieval.

A final direction for future research is the task of complementing the information gathered by applying Web mining techniques with information extracted from the audio signal. Integrating snippets, or even full length versions, of digital music files into AGMIS would certainly increase the value of the

system. However, legal restrictions must be taken into account when heading in this direction. The availability of audio signal-based similarity information on the track level would enable enhanced services and applications, like automatic playlist generation or user interfaces to explore huge music collections in virtual spaces. Bringing AGMIS to the track level would also permit to provide song lyrics since approaches to automatically extracting a correct version of a song's lyrics do already exist. Employing approaches to align audio and lyrics could eventually even yield applications like an automatic karaoke system.

Bibliography

[Alani et al., 2003] Alani, H., Kim, S., Millard, D. E., Weal, M. J., Hall, W., Lewis, P. H., and Shadbolt, N. R. (2003). Automatic Ontology-Based Knowledge Extraction from Web Documents. *IEEE Intelligent Systems*, 18(1):14–21.

[alb, 2008] alb (2008). http://www.unrealvoodoo.org/hiteck/projects/albumart (access: February 2008).

[Allamanche et al., 2003] Allamanche, E., Herre, J., Hellmuth, O., Kastner, T., and Ertel, C. (2003). A Multiple Feature Model for Musical Similarity Retrieval. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR 2003)*, Baltimore, Maryland, USA.

[alt, 2008] alt (2008). http://www.altavista.com (access: February 2008).

[ama, 2008a] ama (2008a). http://amarok.kde.org (access: February 2008).

[ama, 2008b] ama (2008b). http://www.amazon.com (access: January 2008).

[amg, 2007a] amg (2007a). http://www.allmusic.com (access: November 2007).

[amg, 2007b] amg (2007b). http://www.allmusic.com/cg/amg.dll?p=amg&sql=32:amg/info_pages/a_about.html (access: November 2007).

[amg, 2008] amg (2008). http://www.allmusic.com/cg/amg.dll?p=amg&sql=32:amg/info_pages/a_siteglossary.html (access: January 2008).

[Andrews and Heidegger, 1998] Andrews, K. and Heidegger, H. (1998). Information Slices: Visualising and Exploring Large Hierarchies using Cascading, Semi-Circular Discs. In *Proc. of the 4th IEEE Symposium on Information Visualization (InfoVis 1998)*, pages 9–12, Research Triangle Park, NC, USA.

[art, 2008] art (2008). http://www.artofthemix.org (access: February 2008).

[Aucouturier and Pachet, 2003] Aucouturier, J.-J. and Pachet, F. (2003). Representing Musical Genre: A State of the Art. *Journal of New Music Research*, 32(1):83–93.

[Aucouturier and Pachet, 2004] Aucouturier, J.-J. and Pachet, F. (2004). Improving Timbre Similarity: How High is the Sky? *Journal of Negative Results in Speech and Audio Sciences*, 1(1).

[Aucouturier et al., 2007] Aucouturier, J.-J., Pachet, F., Roy, P., and Beurivé, A. (2007). Signal + Context = Better Classification. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR 2007)*, Vienna, Austria.

[Aucouturier et al., 2005] Aucouturier, J.-J., Pachet, F., and Sandler, M. (2005). "The Way It Sounds": Timbre Models for Analysis and Retrieval of Music Signals. *IEEE Transactions on Multimedia*, 7(6):1028–1035.

[Bainbridge, 1997] Bainbridge, D. (1997). *Extensible Optical Music Recognition*. PhD thesis, University of Canterbury, Christchurch, New Zealand.

[Bainbridge et al., 2004] Bainbridge, D., Cunningham, S. J., and Downie, J. S. (2004). Visual Collaging of Music in a Digital Library. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR 2004)*, Barcelona, Spain.

[Bergmark et al., 2002] Bergmark, D., Lagoze, C., and Sbityakov, A. (2002). Focused Crawls, Tunneling, and Digital Libraries. In *Proceedings of the 6th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2002)*, pages 91–106, London, UK. Springer.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Series in Information Sciences and Statistics. Springer.

[Bladh et al., 2004] Bladh, T., Carr, D. A., and Scholl, J. (2004). Extending Tree-Maps to Three Dimensions: A Comparative Study. In *Proceedings of the 6th Asia-Pacific Conference on Computer-Human-Interaction (APCHI 2004)*, Rotorua, New Zealand.

[Brochu et al., 2003] Brochu, E., de Freitas, N., and Bao, K. (2003). The Sound of an Album Cover: Probabilistic Multimedia and IR. In *Proceedings of the 9th International Workshop on Artificial Intelligence and Statistics*, Key West, Florida, USA.

[Bruls et al., 2000] Bruls, M., Huizing, K., and van Wijk, J. J. (2000). Squarified Treemaps. In *Proceedings of the 2nd Joint Eurographics and IEEE TCVG Symposium on Visualization 2000*, pages 33–42, Amsterdam, the Netherlands.

[Burred and Lerch, 2003] Burred, J. J. and Lerch, A. (2003). A Hierarchical Approach to Automatic Musical Genre Classification. In *Proceedings of the 6th International Conference on Digital Audio Effects (DAFx-03)*, London, UK.

[Byrd and Schindele, 2006] Byrd, D. and Schindele, M. (2006). Prospects for Improving OMR with Multiple Recognizers. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006)*, Victoria, Canada.

[Callan and Mitamura, 2002] Callan, J. and Mitamura, T. (2002). Knowledge-Based Extraction of Named Entities. In *Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM 2002)*, pages 532–537, McLean, VA, USA. ACM Press.

[Cano et al., 2002] Cano, P., Kaltenbrunner, M., Gouyon, F., and Batlle, E. (2002). On the Use of Fastmap for Audio Retrieval and Browsing. In *Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR 2002)*, Paris, France.

[Cano and Koppenberger, 2004] Cano, P. and Koppenberger, M. (2004). The Emergence of Complex Network Patterns in Music Artist Networks. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR 2004)*, pages 466–469, Barcelona, Spain.

[Celma et al., 2006] Celma, O., Cano, P., and Herrera, P. (2006). SearchSounds: An Audio Crawler Focused on Weblogs. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006)*, Victoria, Canada.

[Celma and Lamere, 2007] Celma, O. and Lamere, P. (2007). ISMIR 2007 Tutorial: Music Recommendation.
http://mtg.upf.edu/∼ocelma/MusicRecommendationTutorial-ISMIR2007 (access: December 2007).

[Celma et al., 2005] Celma, O., Ramírez, M., and Herrera, P. (2005). Foafing the Music: A Music Recommendation System Based on RSS Feeds and User Preferences. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*, London, UK.

[Chakrabarti, 2002] Chakrabarti, S. (2002). *Mining the Web: Analysis of Hypertext and Semi Structured Data*. Morgan Kaufmann, San Francisco, CA, USA.

[Chakrabarti et al., 1999] Chakrabarti, S., van den Berg, M., and Dom, B. (1999). Focused crawling: A new approach to topic-specific web resource discovery. *Computer Networks*, 31(11–16):1623–1640.

[Cimiano et al., 2004] Cimiano, P., Handschuh, S., and Staab, S. (2004). Towards the Self-Annotating Web. In *Proceedings of the 13th International Conference on World Wide Web (WWW 2004)*, pages 462–471, New York, NY, USA. ACM Press.

[Cimiano and Staab, 2004] Cimiano, P. and Staab, S. (2004). Learning by Googling. *ACM SIGKDD Explorations Newsletter*, 6(2):24–33.

[Cohen and Fan, 2000] Cohen, W. W. and Fan, W. (2000). Web-Collaborative Filtering: Recommending Music by Crawling The Web. *WWW9 / Computer Networks*, 33(1–6):685–698.

[Cohen et al., 2003] Cohen, W. W., Ravikumar, P., and Fienberg, S. E. (2003). A Comparison of String Distance Metrics for Name-Matching Tasks. In *Proceedings of the IJCAI-03 Workshop on Information Integration on the Web (IIWeb-03)*, pages 73–78, Acapulco, Mexico.

[com, 2007] com (2007). http://www.cp.jku.at/CoMIRVA (access: December 2007).

[Corpet, 1988] Corpet, F. (1988). Multiple Sequence Alignment with Hierarchical Clustering. *Nucleic Acids Research*, 16(22):10881–10890.

[cov, 2007] cov (2007). http://www.coveruniverse.com (access: June 2007).

[cov, 2008] cov (2008). http://www.apple.com/itunes/jukebox/coverflow.html (access: February 2008).

[Cox and Cox, 1994] Cox, T. F. and Cox, M. A. A. (1994). *Multidimensional Scaling*. Chapman & Hall.

[Coxeter, 1998] Coxeter, H. S. M. (1998). *Non-Euclidean Geometry*. The Mathematical Association of America, Washington, DC, USA, 6th edition.

[dis, 2007] dis (2007). http://www.discogs.com (access: November 2007).

[Dittenbach et al., 2002] Dittenbach, M., Rauber, A., and Merkl, D. (2002). Uncovering Hierarchical Structure in Data Using the Growing Hierarchical Self-Organizing Map. *Neurocomputing*, 48(1–4):199–216.

[Dixon et al., 2004] Dixon, S., Gouyon, F., and Widmer, G. (2004). Towards Characterisation of Music via Rhythmic Patterns. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR 2004)*, pages 509–516, Barcelona, Spain.

[Dixon et al., 2003] Dixon, S., Pampalk, E., and Widmer, G. (2003). Classification of Dance Music by Periodicity Patterns. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR 2003)*, pages 159–166, Baltimore, Maryland, USA. John Hopkins University.

[Eades, 1984] Eades, P. A. (1984). A Heuristic for Graph Drawing. In *Congressus Numerantium*, volume 42, pages 149–160.

[ebay, 2007] ebay (2007). http://www.ebay.com (access: October 2007).

[Eck et al., 2007] Eck, D., Bertin-Mahieux, T., and Lamere, P. (2007). Autotagging Music Using Supervised Machine Learning. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR 2007)*, Vienna, Austria.

[Elias, 1975] Elias, P. (1975). Universal Codeword Sets and Representations of the Integers. *IEEE Transactions on Information Theory*, 21(2):194–203.

[Ellis et al., 2002] Ellis, D. P., Whitman, B., Berenzweig, A., and Lawrence, S. (2002). The Quest For Ground Truth in Musical Artist Similarity. In *Proceedings of 3rd International Conference on Music Information Retrieval (ISMIR 2002)*, Paris, France.

[epi, 2007] epi (2007). http://www.epinions.com/music (access: August 2007).

[Eppstein et al., 1997] Eppstein, D., Paterson, M., and Yao, F. F. (1997). On Nearest-Neighbor Graphs. *Discrete & Computational Geometry*, 17(3):263–282.

[Erdős and Rényi, 1959] Erdős, P. and Rényi, A. (1959). On Random Graphs. *Publications Mathematicae*, 6:290–297.

[evi, 2007] evi (2007). http://www.evillabs.sk/evillyrics (access: August 2007).

[exa, 2007] exa (2007). http://www.exalead.com (access: August 2007).

[Feng et al., 2003] Feng, D. D., Siu, W.-C., and Zhang, H.-J. (2003). *Multimedia Information Retrieval and Management: Technological Fundamentals and Applications*, pages 5–6. Springer.

[Foote, 1997] Foote, J. T. (1997). Content-Based Retrieval of Music and Audio. In Kuo, C., editor, *Proceedings of SPIE Multimedia Storage and Archiving Systems II*, volume 3229, pages 138–147.

[fre, 2008] fre (2008). http://www.freedb.org (access: February 2008).

[Friedman, 1940] Friedman, M. (1940). A Comparison of Alternative Tests of Significance for the Problem of m Rankings. *The Annals of Mathematical Statistics*, 11(1):86–92.

[Fruchterman and Reingold, 1991] Fruchterman, T. M. J. and Reingold, E. M. (1991). Graph Drawing by Force-directed Placement. *Software – Practice and Experience*, 21(11):1129–1164.

[Fürnkranz, 2002] Fürnkranz, J. (2002). Web Structure Mining – Exploiting the Graph Structure of the World-Wide Web. *ÖGAI-Journal*, 21(2):17–26.

[Geleijnse and Korst, 2006] Geleijnse, G. and Korst, J. (2006). Web-based Artist Categorization. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006)*, Victoria, Canada.

[Geleijnse et al., 2007] Geleijnse, G., Schedl, M., and Knees, P. (2007). The Quest for Ground Truth in Musical Artist Tagging in the Social Web Era. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR 2007)*, Vienna, Austria.

[Golomb, 1966] Golomb, S. W. (1966). Run-Length Encodings. *IEEE Transactions on Information Theory*, 12:399–401.

[Gómez, 2006] Gómez, E. (2006). *Tonal Description of Music Audio Signals*. PhD thesis, Universitat Pompeu Fabra, Barcelona, Spain.

[goop, 2007] goop (2007). http://www.google.com/intl/en/options (access: November 2007).

[Gouyon et al., 2004] Gouyon, F., Dixon, S., Pampalk, E., and Widmer, G. (2004). Evaluating Rhythmic Descriptors for Musical Genre Classification. In *Proceedings of the 25th AES International Conference*, London, UK.

[gra, 2008] gra (2008). http://www.gracenote.com (access: February 2008).

[Harris, 1999] Harris, R. L. (1999). *Information Graphics: A Comprehensive Illustrated Reference: Visual Tools for Analyzing, Managing, and Communicating*. Oxford University Press, New York, NY, USA.

[Hearst, 1992] Hearst, M. A. (1992). Automatic Acquisition of Hyponyms from Large Text Corpora. In *Proceedings of the 14th Conference on Computational Linguistics – Vol. 2*, pages 539–545, Nantes, France.

[Hotelling, 1933] Hotelling, H. (1933). Analysis of a Complex of Statistical Variables Into Principal Components. *Journal of Educational Psychology*, 24:417–441 and 498–520.

[Hu et al., 2007] Hu, X., Bay, M., and Downie, J. S. (2007). Creating a Simplified Music Mood Classification Ground-Truth Set. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR 2007)*, Vienna, Austria.

[Hu et al., 2006] Hu, X., Downie, J. S., and Ehmann, A. (2006). Exploiting Recommended Usage Metadata: Exploratory Analyses. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006)*, Victoria, Canada.

[Hu et al., 2005] Hu, X., Downie, J. S., West, K., and Ehmann, A. (2005). Mining Music Reviews: Promising Preliminary Results. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*, London, UK.

[hyp, 2008] hyp (2008). http://hypertree.woot.com.ar (access: May 2008).

[ism, 2007] ism (2007). http://www.ismir.net/proceedings (access: December 2007).

[isp, 2006] isp (2006). http://wordlist.sourceforge.net (access: June 2006).

[Johnson and Shneiderman, 1991] Johnson, B. and Shneiderman, B. (1991). Tree-Maps: A Space-Filling Approach to the Visualization of Hierarchical Information Structures. In *Proceedings of the 2nd IEEE Conference on Visualization 1991 (Vis 1991)*, pages 284–291, San Diego, CA, USA.

[Keim et al., 2005] Keim, D. A., Schneidewind, J., and Sips, M. (2005). Interactive Poster: FP-Viz: Visual Pattern Mining. In *Proceedings of IEEE Information Visualization 2005 (InfoVis 2005)*, Minneapolis, Minnesota, USA.

[Kirkpatrick et al., 1983] Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220(4598):671–680.

[Knees et al., 2004] Knees, P., Pampalk, E., and Widmer, G. (2004). Artist Classification with Web-based Data. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR 2004)*, pages 517–524, Barcelona, Spain.

[Knees et al., 2006a] Knees, P., Pohle, T., Schedl, M., and Widmer, G. (2006a). Combining Audio-based Similarity with Web-based Data to Accelerate Automatic Music Playlist Generation. In *Proceedings of the 8th ACM SIGMM International Workshop on Multimedia Information Retrieval (MIR'06)*, Santa Barbara, CA, USA.

[Knees et al., 2007a] Knees, P., Pohle, T., Schedl, M., and Widmer, G. (2007a). A Music Search Engine Built upon Audio-based and Web-based Similarity Measures. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2007)*, Amsterdam, the Netherlands.

[Knees et al., 2006b] Knees, P., Schedl, M., Pohle, T., and Widmer, G. (2006b). An Innovative Three-Dimensional User Interface for Exploring Music Collections Enriched with Meta-Information from the Web. In *Proceedings of the 14th ACM International Conference on Multimedia (MM 2006)*, Santa Barbara, CA, USA.

[Knees et al., 2007b] Knees, P., Schedl, M., Pohle, T., and Widmer, G. (2007b). Exploring Music Collections in Virtual Landscapes. *IEEE MultiMedia*, 14(3):46–54.

[Knees et al., 2005] Knees, P., Schedl, M., and Widmer, G. (2005). Multiple Lyrics Alignment: Automatic Retrieval of Song Lyrics. In *Proceedings of 6th International Conference on Music Information Retrieval (ISMIR 2005)*, pages 564–569, London, UK.

[Kobsa, 2004] Kobsa, A. (2004). User Experiments with Tree Visualization Systems. In *Proceedings of the 10th IEEE Symposium on Information Visualization 2004 (InfoVis 2004)*, Austin, Texas, USA.

[Kohonen, 1982] Kohonen, T. (1982). Self-Organizing Formation of Topologically Correct Feature Maps. *Biological Cybernetics*, 43:59–69.

[Kohonen, 2001] Kohonen, T. (2001). *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Germany, 3rd edition.

[Korst and Geleijnse, 2006] Korst, J. and Geleijnse, G. (2006). Efficient lyrics retrieval and alignment. In Verhaegh, W., Aarts, E., ten Kate, W., Korst, J., and Pauws, S., editors, *Proceedings of the 3rd Philips Symposium on Intelligent Algorithms (SOIA 2006)*, pages 205–218, Eindhoven, the Netherlands.

[Kruskal and Wish, 1978] Kruskal, J. B. and Wish, M. (1978). *Multidimensional Scaling*. Paper Series on Quantitative Applications in the Social Sciences. Sage Publications, Newbury Park, CA, USA.

[Lamping and Rao, 1996] Lamping, J. and Rao, R. (1996). Visualizing Large Trees Using the Hyperbolic Browser. In *CHI'96: Conference Companion on Human Factors in Computing Systems*, pages 388–389, Vancouver, Canada.

[Lamping et al., 1995] Lamping, J., Rao, R., and Pirolli, P. (1995). A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies. In *CHI'95: Proceedings of the ACM SIGCHI Conference on Human factors in Computing Systems*, pages 401–408, Denver, CO, USA.

[las, 2007] las (2007). http://last.fm (access: December 2007).

[Lawler et al., 1985] Lawler, E. L., Lenstra, J. K., Kan, A. H. G. R., and Shmoys, D. B. (1985). *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. Wiley Series in Discrete Mathematics and Optimization. Wiley.

[Lee and Seung, 1999] Lee, D. D. and Seung, H. S. (1999). Learning the Parts of Objects by Non-negative Matrix Factorization. *Nature*, 401(6755):788–791.

[leo, 2007] leo (2007). http://dict.leo.org (access: May 2007).

[Liu, 2007] Liu, B. (2007). *Web Data Mining – Exploring Hyperlinks, Contents and Usage Data*. Springer, Berlin, Heidelberg, Germany.

[Logan, 2000] Logan, B. (2000). Mel Frequency Cepstral Coefficients for Music Modeling. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR 2000)*, Plymouth, Massachusetts, USA.

[Logan, 2002] Logan, B. (2002). Content-based Playlist Generation: Exploratory Experiments. In *Proceedings of the 3rd International Symposium on Music Information Retrieval (ISMIR 2002)*, pages 295–296, Paris, France.

[Logan et al., 2004] Logan, B., Kositsky, A., and Moreno, P. (2004). Semantic Analysis of Song Lyrics. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2004)*, Taipei, Taiwan.

[Logan and Salomon, 2001] Logan, B. and Salomon, A. (2001). A Music Similarity Function Based on Signal Analysis. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2001)*, Tokyo, Japan. Institute of Electrical and Electronics Engineers.

[luc, 2008] luc (2008). http://lucene.apache.org (access: January 2008).

[Luhn, 1957] Luhn, H. P. (1957). A Statistical Approach to Mechanized Encoding and Searching of Literary Information. *IBM Journal*, pages 309–317.

[Mahedero et al., 2005] Mahedero, J. P. G., Martínez, A., Cano, P., Koppenberger, M., and Gouyon, F. (2005). Natural language processing of lyrics. In *Proceedings of the 13th ACM International Conference on Multimedia (MM 2005)*, pages 475–478, Singapore, Singapore.

[Mandel and Ellis, 2005] Mandel, M. I. and Ellis, D. P. (2005). Song-Level Features and Support Vector Machines for Music Classification. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*, London, UK.

[McKay and Fujinaga, 2004] McKay, C. and Fujinaga, I. (2004). Automatic Genre Classification Using Large High-Level Musical Feature Sets. In *Proceedings of 5th International Conference on Music Information Retrieval (ISMIR 2004)*, pages 525–530, Barcelona, Spain.

[Mobasher, 2004] Mobasher, B. (2004). *Practical Handbook of Internet Computing*, chapter Web Usage Mining and Personalization. CRC Press.

[Moore, 2001] Moore, A. (2001). Categorical Conventions in Music Discourse: Style and Genre. *Music & Letters*, 82(3):432–442.

[mp3, 2008] mp3 (2008). http://www.mp3.com (access: January 2008).

[msn, 2007] msn (2007). http://music.msn.com (access: November 2007).

[mtr, 2008] mtr (2008). http://www.musictrails.com.ar (access: April 2008).

[mus, 2008] mus (2008). http://www.musicplasma.com (access: February 2008).

[mys, 2008] mys (2008). http://www.mysql.com (access: June 2008).

[nor, 2008] nor (2008). http://www.northernlight.com (access: February 2008).

[Ong, 2005] Ong, B. S. (2005). *Towards Automatic Music Structural Analysis: Identifying Characteristic Within-Song Excerpts in Popular Music*. PhD thesis, Universitat Pompeu Fabra, Barcelona, Spain.

[Pachet and Cazaly, 2000] Pachet, F. and Cazaly, D. (2000). A Taxonomy of Musical Genre. In *Proceedings of Content-Based Multimedia Information Access (RIAO) Conference*, Paris, France.

[Pachet and Roy, 2007] Pachet, F. and Roy, P. (2007). Exploring Billions of Audio Features. In *Proceedings of the 5th International Workshop on Content-Based Multimedia Indexing (CBMI 2007)*, pages 227–235, Bordeaux, France.

[Pachet et al., 2001] Pachet, F., Westerman, G., and Laigre, D. (2001). Musical Data Mining for Electronic Music Distribution. In *Proceedings of the 1st International Conference on Web Delivering of Music (WEDELMUSIC 2001)*, Florence, Italy.

[Page et al., 1998] Page, L., Brin, S., Motwani, R., and Winograd, T. (1998). The PageRank Citation Ranking: Bringing Order to the Web. In *Proceedings of the Annual Meeting of the American Society for Information Science (ASIS 1998)*, pages 161–172.

[Pampalk, 2006] Pampalk, E. (2006). *Computational Models of Music Similarity and their Application to Music Information Retrieval*. PhD thesis, Vienna University of Technology.

[Pampalk et al., 2004] Pampalk, E., Dixon, S., and Widmer, G. (2004). Exploring Music Collections by Browsing Different Views. *Computer Music Journal*, 28(3).

[Pampalk et al., 2005] Pampalk, E., Flexer, A., and Widmer, G. (2005). Hierarchical Organization and Description of Music Collections at the Artist Level. In *Proceedings of the 9th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2005)*, Vienna, Austria.

[Pampalk et al., 2002a] Pampalk, E., Rauber, A., and Merkl, D. (2002a). Content-based Organization and Visualization of Music Archives. In *Proceedings of the 10th ACM International Conference on Multimedia (MM 2002)*, pages 570–579, Juan les Pins, France.

[Pampalk et al., 2002b] Pampalk, E., Rauber, A., and Merkl, D. (2002b). Using Smoothed Data Histograms for Cluster Visualization in Self-Organizing Maps. In *Proceedings of the International Conference on Artificial Neural Networks (ICANN 2002)*, pages 871–876, Madrid, Spain. Springer.

[Pohle, 2005] Pohle, T. (2005). Extraction of Audio Descriptors and their Evaluation in Music Classification Tasks. Master's thesis, Technische Universität Kaiserslautern, Austrian Research Institute for Artificial Intelligence (ÖFAI), Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI).

[Pohle et al., 2007a] Pohle, T., Knees, P., Schedl, M., Pampalk, E., and Widmer, G. (2007a). "Reinventing the Wheel": A Novel Approach to Music Player Interfaces. *IEEE Transactions on Multimedia*, 9:567–575.

[Pohle et al., 2007b] Pohle, T., Knees, P., Schedl, M., and Widmer, G. (2007b). Building an Interactive Next-Generation Artist Recommender Based on Automatically Derived High-Level Concepts. In *Proceedings of the 5th International Workshop on Content-Based Multimedia Indexing (CBMI'07)*, Bordeaux, France.

[Pohle et al., 2005] Pohle, T., Pampalk, E., and Widmer, G. (2005). Generating Similarity-based Playlists Using Traveling Salesman Algorithms. In *Proceedings of the 8th International Conference on Digital Audio Effects (DAFx-05)*, pages 220–225, Madrid, Spain.

[Pohle and Schnitzer, 2007] Pohle, T. and Schnitzer, D. (2007). Striving for an Improved Audio Similarity Measure. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR 2007)*, Vienna, Austria.

[Pölzlbauer et al., 2005a] Pölzlbauer, G., Rauber, A., and Dittenbach, M. (2005a). A Vector Field Visualization Technique for Self-Organizing Maps. In Tu Bao Ho, David Cheung, H. L., editor, *Proceedings of the 9th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2005)*, pages 399–409, Hanoi, Vietnam. Springer.

[Pölzlbauer et al., 2005b] Pölzlbauer, G., Rauber, A., and Dittenbach, M. (2005b). Advanced Visualization Techniques for Self-Organizing Maps with Graph-Based Methods. In Jun Wang, Xiaofeng Liao, Z. Y., editor, *Proceedings of the 2nd International Symposium on Neural Networks (ISNN 2005)*, pages 75–80, Chongqing, China. Springer.

[Porter, 1980] Porter, M. F. (1980). An Algorithm for Suffix Stripping. *Program*, 13(3):130–137.

[Porter, 1997] Porter, M. F. (1997). *An Algorithm for Suffix Stripping*, pages 313–316. Morgan Kaufmann, San Francisco, CA, USA.

[pro, 2007] pro (2007). http://www.processing.org (access: March 2007).

[Robertson and Jones, 1988] Robertson, S. E. and Jones, K. S. (1988). *Relevance Weighting of Search Terms*, pages 143–160. Taylor Graham Publishing, London, UK.

[Russell and Norvig, 2003a] Russell, S. J. and Norvig, P. (2003a). *Artificial Intelligence: A Modern Approach*, pages 95–97. Prentice Hall, Englewood Cliffs, NJ, USA, 2nd edition.

[Russell and Norvig, 2003b] Russell, S. J. and Norvig, P. (2003b). *Artificial Intelligence: A Modern Approach*, pages 73–74. Prentice Hall, Englewood Cliffs, NJ, USA, 2nd edition.

[Salton, 1962] Salton, G. (1962). The Use of Citations as an Aid to Automatic Content Analysis. Technical Report ISR-2, Section III, Harvard Computation Laboratory, Cambridge, MA, USA.

[Salton and Buckley, 1988] Salton, G. and Buckley, C. (1988). Term-weighting Approaches in Automatic Text Retrieval. *Information Processing and Management*, 24(5):513–523.

[Salton et al., 1975] Salton, G., Wong, A., and Yang, C. S. (1975). A Vector Space Model for Automatic Indexing. *Communications of the ACM*, 18(11):613–620.

[Sammon, 1969] Sammon, J. W. (1969). A Nonlinear Mapping for Data Structure Analysis. *IEEE Transactions on Computers*, 18:401–409.

[Schedl, 2003] Schedl, M. (2003). An Explorative, Hierarchical User Interface to Structured Music Repositories. Master's thesis, Vienna University of Technology, Vienna, Austria.

[Schedl et al., 2006a] Schedl, M., Knees, P., Pohle, T., and Widmer, G. (2006a). Towards Automatic Retrieval of Album Covers. In *Proceedings of the 28th European Conference on Information Retrieval (ECIR 2006)*, London, UK.

[Schedl et al., 2008] Schedl, M., Knees, P., Pohle, T., and Widmer, G. (2008). Towards an Automatically Generated Music Information System via Web Content Mining. In *Proceedings of the 30th European Conference on Information Retrieval (ECIR 2008)*, Glasgow, Scotland.

[Schedl et al., 2007a] Schedl, M., Knees, P., Seyerlehner, K., and Pohle, T. (2007a). The CoMIRVA Toolkit for Visualizing Music-Related Data. In *Proceedings of the 9th Eurographics/IEEE VGTC Symposium on Visualization (EuroVis 2007)*, Norrköping, Sweden.

[Schedl et al., 2005a] Schedl, M., Knees, P., and Widmer, G. (2005a). A Web-Based Approach to Assessing Artist Similarity using Co-Occurrences. In *Proceedings of the 4th International Workshop on Content-Based Multimedia Indexing (CBMI 2005)*, Riga, Latvia.

[Schedl et al., 2005b] Schedl, M., Knees, P., and Widmer, G. (2005b). Discovering and Visualizing Prototypical Artists by Web-based Co-Occurrence Analysis. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*, London, UK.

[Schedl et al., 2005c] Schedl, M., Knees, P., and Widmer, G. (2005c). Improving Prototypical Artist Detection by Penalizing Exorbitant Popularity. In *Proceedings of the 3rd International Symposium on Computer Music Modeling and Retrieval (CMMR 2005)*, Pisa, Italy.

[Schedl et al., 2005d] Schedl, M., Knees, P., and Widmer, G. (2005d). Interactive Poster: Using CoMIRVA for Visualizing Similarities Between Music Artists. In *Proceedings of the 16th IEEE Visualization 2005 Conference (Vis 2005)*, Minneapolis, Minnesota, USA.

[Schedl et al., 2006b] Schedl, M., Knees, P., and Widmer, G. (2006b). Investigating Web-Based Approaches to Revealing Prototypical Music Artists in Genre Taxonomies. In *Proceedings of the 1st IEEE International Conference on Digital Information Management (ICDIM 2006)*, Bangalore, India.

[Schedl et al., 2007b] Schedl, M., Knees, P., Widmer, G., Seyerlehner, K., and Pohle, T. (2007b). Browsing the Web Using Stacked Three-Dimensional Sunbursts to Visualize Term Co-Occurrences and Multimedia Content. In *Proceedings of the 18th IEEE Visualization 2007 Conference (Vis 2007)*, Sacramento, CA, USA.

[Schedl et al., 2006c] Schedl, M., Pohle, T., Knees, P., and Widmer, G. (2006c). Assigning and Visualizing Music Genres by Web-based Co-Occurrence Analysis. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006)*, Victoria, Canada.

[Schedl and Widmer, 2007] Schedl, M. and Widmer, G. (2007). Automatically Detecting Members and Instrumentation of Music Bands via Web Content Mining. In *Proceedings of the 5th Workshop on Adaptive Multimedia Retrieval (AMR 2007)*, Paris, France.

[Schedl et al., 2007c] Schedl, M., Widmer, G., Pohle, T., and Seyerlehner, K. (2007c). Web-based Detection of Music Band Members and Line-Up. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR 2007)*, Vienna, Austria.

[Scheirer and Slaney, 1997] Scheirer, E. and Slaney, M. (1997). Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP 1997)*, pages 1331–1334, Munich, Germany.

[sea, 2008] sea (2008). http://www.searchsounds.net (access: February 2008).

[sec, 2008] sec (2008). http://secondstring.sourceforge.net (access: June 2008).

[Seyerlehner, 2006] Seyerlehner, K. (2006). Inhaltsbasierte Ähnlichkeitsmetriken zur Navigation in Musiksammlungen. Master's thesis, Johannes Kepler Universität Linz, Linz, Austria.

[Seyerlehner et al., 2007] Seyerlehner, K., Pohle, T., Schedl, M., and Widmer, G. (2007). Automatic Music Detection in Television Productions. In *Proceedings of the 10th International Conference on Digital Audio Effects (DAFx-07)*, Bordeaux, France.

[Sheskin, 2004] Sheskin, D. J. (2004). *Handbook of Parametric and Nonparametric Statistical Procedures*. Chapman & Hall/CRC, Boca Raton, London, New York, Washington, DC, 3rd edition.

[Shneiderman, 1992] Shneiderman, B. (1992). Tree Visualization with Tree-Maps: 2-D Space-Filling Approach. *ACM Transactions on Graphics*, 11(1):92–99.

[Skiena, 1997] Skiena, S. S. (1997). *The Algorithm Design Manual*. Springer, Department of Computer Science, State University of New York, Stony Brook, NY 11794-4400.

[sno, 2008] sno (2008). http://snowball.tartarus.org (access: mai 2008).

[Spence, 2007] Spence, R. (2007). *Information Visualization – Design for Interaction*. Pearson, Prentice Hall, Harlow, England, 2nd edition.

[Srivastava et al., 2000] Srivastava, J., Cooley, R., Deshpande, M., and Tan, P.-N. (2000). Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data. *ACM SIGKDD Explorations Newsletter*, 1(2):12–23.

[Stasko et al., 2000] Stasko, J., Catrambone, R., Guzdial, M., and McDonald, K. (2000). An Evaluation of Space-Filling Information Visualizations for Depicting Hierarchical Structures. *International Journal of Human-Computer Studies*, 53(5):663–694.

[Stasko and Zhang, 2000] Stasko, J. and Zhang, E. (2000). Focus+Context Display and Navigation Techniques for Enhancing Radial, Space-Filling Hierarchy Visualizations. In *Proceedings of the 6th IEEE Symposium on Information Visualization (InfoVis 2000)*, pages 57–65, Salt Lake City, UT, USA.

[sto, 2008] sto (2008). http://www.ranks.nl/stopwords (access: mai 2008).

[the, 2007] the (2007). http://www.thesaurus.com (access: May 2007).

[Tsymbalenko and Munson, 2001] Tsymbalenko, Y. and Munson, E. V. (2001). Using HTML Metadata to Find Relevant Images on the World Wide Web. *Internet Computing 2001*, pages 842–848.

[Tufte, 2001] Tufte, E. R. (2001). *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, CT, USA.

[Tzanetakis and Cook, 2002] Tzanetakis, G. and Cook, P. (2002). Musical Genre Classification of Audio Signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302.

[Ultsch and Siemon, 1990] Ultsch, A. and Siemon, H. P. (1990). Kohonen's Self-Organizing Feature Maps for Exploratory Data Analysis. In *Proceedings of the International Neural Network Conference (INNC 1990)*, pages 305–308, Dordrecht, the Netherlands. Kluwer Academic Publishers.

[van Laarhoven and Aarts, 1987] van Laarhoven, P. J. M. and Aarts, E. H. L., editors (1987). *Simulated Annealing: Theory and Applications*. Kluwer Academic Publishers, Norwell, MA, USA.

[van Rijsbergen, 1979] van Rijsbergen, C. J. (1979). *Information Retrieval*. Butterworths, London, UK, 2nd edition.

[van Wijk and van de Wetering, 1999] van Wijk, J. J. and van de Wetering, H. (1999). Cushion Treemaps: Visualization of Hierarchical Information. In *Proceedings of the 5th IEEE Symposium on Information Visualization 1999 (InfoVis 1999)*, pages 73–78, San Francisco, CA, USA.

[Vesanto, 1999] Vesanto, J. (1999). SOM-Based Data Visualization Methods. *Intelligent Data Analysis*, 3(2):111–126.

[Vesanto, 2002] Vesanto, J. (2002). *Data Exploration Process Based on the Self-Organizing Map*. PhD thesis, Helsinki University of Technology, Espoo, Finland.

[Vignoli et al., 2004] Vignoli, F., van Gulik, R., and van de Wetering, H. (2004). Mapping Music in the Palm of Your Hand, Explore and Discover Your Collection. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR 2004)*, Barcelona, Spain.

[Walter and Ritter, 2002] Walter, J. A. and Ritter, H. (2002). On interactive visualization of high-dimensional data using the hyperbolic plane. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 123–131, Edmonton, Canada. ACM Press.

[Ware, 2004] Ware, C. (2004). *Information Visualization: Perception for Design*. Morgan Kaufmann, San Francisco, CA, USA.

[web, 2007] web (2007). http://www.cs.uic.edu/∼liub/WebContentMining.html (access: December 2007).

[wgt, 2007] wgt (2007). http://www.gnu.org/software/wget (access: March 2007).

[Whitman and Lawrence, 2002] Whitman, B. and Lawrence, S. (2002). Inferring Descriptions and Similarity for Music from Community Metadata. In *Proceedings of the 2002 International Computer Music Conference (ICMC 2002)*, pages 591–598, Göteborg, Sweden.

[Whitman and Smaragdis, 2002] Whitman, B. and Smaragdis, P. (2002). Combining Musical and Cultural Features for Intelligent Style Detection. In *Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR 2002)*, pages 47–52, Paris, France.

[wik, 2007a] wik (2007a). http://en.wikipedia.org/wiki/List_of_file_formats (access: March 2007).

[wik, 2007b] wik (2007b). http://www.wikipedia.org (access: December 2007).

[Xu et al., 2003a] Xu, C., Maddage, N. C., Shao, X., and Tian, Q. (2003a). Musical Genre Classification Using Support Vector Machines. In *Proceedings of the International Conference of Acoustics, Speech, and Signal Processing (ICASSP 2003)*, Hong Kong, China.

[Xu and Zuo, 2007] Xu, Q. and Zuo, W. (2007). First-order Focused Crawling. In *Proceedings of the 16th International Conference on World Wide Web (WWW 2007)*, pages 1159–1160, Banff, Canada. ACM.

[Xu et al., 2003b] Xu, W., Liu, X., and Gong, Y. (2003b). Document Clustering Based on Non-negative Matrix Factorization. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2003)*, pages 267–273, Toronto, Canada. ACM Press.

[yah, 2007a] yah (2007a). http://dir.yahoo.com/Entertainment/Music/Genres (access: March 2007).

[yah, 2007b] yah (2007b). http://music.yahoo.com (access: November 2007).

[Yang et al., 2002] Yang, J., Ward, M. O., and Rundensteiner, E. A. (2002). InterRing: An Interactive Tool for Visually Navigating and Manipulating Hierarchical Structures. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis 2002)*, Boston, Massachusetts, USA.

[Zadel and Fujinaga, 2004] Zadel, M. and Fujinaga, I. (2004). Web Services for Music Information Retrieval. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR 2004)*, Barcelona, Spain.

[Zobel and Moffat, 1998] Zobel, J. and Moffat, A. (1998). Exploring the Similarity Space. *ACM SIGIR Forum*, 32(1):18–34.

[Zobel and Moffat, 2006] Zobel, J. and Moffat, A. (2006). Inverted Files for Text Search Engines. *ACM Computing Surveys*, 38:1–56.

Appendix A: Test Collections

Country			
Johnny Cash	Willie Nelson	Dolly Parton	Hank Williams
Faith Hill	Dixie Chicks	Garth Brooks	Kenny Rogers
Tim McGraw	Hank Snow	Brooks and Dunn	Lee Hazlewood
Kenny Chesney	Jim Reeves	Roger Miller	Kris Kristofferson
Folk			
Bob Dylan	Joni Mitchell	Leonard Cohen	Joan Baez
Townes van Zandt	Pete Seeger	Suzanne Vega	Tracy Chapman
Tim Buckley	Steeleye Span	Woodie Guthrie	Donovan
Cat Stevens	John Denver	Don McLean	Crosby Stills & Nash
Jazz			
Miles Davis	Dave Brubeck	Billie Holiday	Duke Ellington
Django Reinhardt	Glenn Miller	Ella Fitzgerald	Louis Armstrong
Nat King Cole	Herbie Hancock	Nina Simone	John Coltrane
Charlie Parker	Count Basie	Thelonious Monk	Cannonball Adderley
Blues			
John Lee Hooker	Muddy Waters	Taj Mahal	John Mayall
Big Bill Broonzy	BB King	Howlin' Wolf	Willie Dixon
Blind Lemon Jefferson	Blind Willie McTell	Mississippi John Hurt	T-Bone Walker
Etta James	Lightnin' Hopkins	Otis Rush	Albert King
RnB / Soul			
James Brown	Marvin Gaye	Otis Redding	Solomon Burke
Sam Cooke	Aretha Franklin	Al Green	The Temptations
The Drifters	Fats Domino	The Supremes	Isaac Hayes
Alicia Keys	Erykah Badu	India Arie	Jill Scott
Heavy Metal / Hard Rock			
Iron Maiden	Megadeth	Slayer	Sepultura
Black Sabbath	Anthrax	Alice Cooper	Deep Purple
Def Leppard	AC/DC	Judas Priest	Kiss
Metallica	Pantera	Queensryche	Skid Row
Alternative Rock / Indie			
Nirvana	Beck	Smashing Pumpkins	Radiohead
Belle and Sebastian	Alice in Chains	Echo and the Bunnymen	Sonic Youth
Weezer	Pearl Jam	Foo Fighters	Hole
Bush	The Smiths	Depeche Mode	Jane's Addiction

Table A-1: Composition of collection *C224a14g*.

Punk			
Patti Smith	Sex Pistols	Sid Vicious	Ramones
Bad Religion	The Clash	NoFX	Dead Kennedys
Buzzcocks	Green Day	Blink 182	Sum 41
The Misfits	Rancid	Screeching Weasel	Pennywise
Rap / Hip-Hop			
Eminem	Dr. Dre	Public Enemy	Missy Elliot
Cypress Hill	50 Cent	Run DMC	Grandmaster Flash
2Pac	Snoop Dogg	Jay-Z	Busta Rhymes
LL Cool J	DMX	Ice Cube	Mystikal
Electronica			
Aphex Twin	Daft Punk	Kraftwerk	Chemical Brothers
Fatboy Slim	Basement Jaxx	Carl Cox	Moloko
Paul Oakenfold	Prodigy	Armand van Helden	Moby
Massive Attack	Mouse on Mars	Jimi Tenor	Underworld
Reggae			
Bob Marley	Jimmy Cliff	Peter Tosh	Ziggy Marley
Sean Paul	Alpha Blondie	Shaggy	Maxi Priest
Shabba Ranks	UB40	Inner Circle	Desmond Dekker
Capleton	Bounty Killer	Eddy Grant	Black Uhuru
Rock 'n' Roll			
The Rolling Stones	The Animals	The Faces	The Kinks
The Who	Elvis Presley	Chuck Berry	Little Richard
Jerry Lee Lewis	Buddy Holly	Bo Diddley	Bill Haley
Chubby Checker	The Yardbirds	Carl Perkins	Gene Vincent
Pop			
Madonna	Britney Spears	N'Sync	Justin Timberlake
ABBA	Michael Jackson	Janet Jackson	Prince
Spice Girls	Christina Aguilera	Robbie Williams	Nelly Furtado
Avril Lavigne	Jennifer Lopez	O-Town	Shakira
Classical			
Wolfgang Amadeus Mozart	Ludwig van Beethoven	Johann Sebastian Bach	Joseph Haydn
Johannes Brahms	Frederic Chopin	Antonin Dvorak	Gustav Mahler
Franz Schubert	Antonio Vivaldi	Richard Wagner	Herbert von Karajan
Yehudi Menuhin	Georg Friedrich Händel	Tchaikovsky	Giuseppe Verdi

Table A-2: Continuation of Table A-1.

Country			
Johnny Cash	Willie Nelson	Dolly Parton	Hank Williams
Faith Hill	Dixie Chicks	Garth Brooks	Kenny Rogers
Folk			
Bob Dylan	Joni Mitchell	Leonard Cohen	Joan Baez
Townes van Zandt	Pete Seeger	Suzanne Vega	Tracy Chapman
Jazz			
Miles Davis	Dave Brubeck	Billie Holiday	Duke Ellington
Django Reinhardt	Glenn Miller	Ella Fitzgerald	Louis Armstrong
Blues			
John Lee Hooker	Muddy Waters	Taj Mahal	John Mayall
Big Bill Broonzy	BB King	Howlin' Wolf	Willie Dixon
RnB / Soul			
James Brown	Marvin Gaye	Otis Redding	Solomon Burke
Sam Cooke	Aretha Franklin	Al Green	The Temptations
Heavy Metal / Hard Rock			
Iron Maiden	Megadeth	Slayer	Sepultura
Black Sabbath	Anthrax	Alice Cooper	Deep Purple
Alternative Rock / Indie			
Nirvana	Beck	Smashing Pumpkins	Radiohead
Belle and Sebastian	Alice in Chains	Echo and the Bunnymen	Sonic Youth
Punk			
Patti Smith	Sex Pistols	Sid Vicious	Ramones
Bad Religion	The Clash	NoFX	Dead Kennedys
Rap / Hip-Hop			
Eminem	Dr. Dre	Public Enemy	Missy Elliot
Cypress Hill	50 Cent	Run DMC	Grandmaster Flash
Electronica			
Aphex Twin	Daft Punk	Kraftwerk	Chemical Brothers
Fatboy Slim	Basement Jaxx	Carl Cox	Moloko
Reggae			
Bob Marley	Jimmy Cliff	Peter Tosh	Ziggy Marley
Sean Paul	Alpha Blondie	Shaggy	Maxi Priest
Rock 'n' Roll			
The Rolling Stones	The Animals	The Faces	The Kinks
The Who	Elvis Presley	Chuck Berry	Little Richard
Pop			
Madonna	Britney Spears	N'Sync	Justin Timberlake
ABBA	Michael Jackson	Janet Jackson	Prince
Classical			
Wolfgang Amadeus Mozart	Ludwig van Beethoven	Johann Sebastian Bach	Joseph Haydn
Johannes Brahms	Frederic Chopin	Antonin Dvorak	Gustav Mahler

Table A-3: Composition of collection *C112a14g*.

A Cappella		
Golden Gate Quartet	Ladysmith Black Mambazo	The Cafe of the Gate of Salvation
The Heavenly Light Quartet		
Acid Jazz		
Count Basic	Jazzanova	Saint Germain
Us3		
Blues		
Etta James	John Lee Hooker	Muddy Waters
Willie Dixon		
Bossa Nova		
Antonio Carlos Jobim	Baden Powell	João Gilberto
Stan Getz, João & Astrud Gilberto		
Celtic		
Clannad	Enya	Lunasa
Sharon Shannon	The Chieftains	
Death Metal		
Borknagar	Cannibal Corpse	Dimmu Borgir
Entombed		
DnB		
DJ Zinc	Dillinja	Ed Rush & Optical
Goldie	Grooverider	
Downtempo		
DJ Shadow	Kruder & Dorfmeister	Massive Attack
Sofa Surfers		
Electronic		
Aphex Twin	Kaito	Ken Ishii
Matthew Herbert		
Euro-Dance		
2 Unlimited	Ace Of Base	Culture Beat
Magic Affair	Masterboy	U96
Folk Rock		
Corvus Corax	In Extremo	Schandmaul
Subway To Sally	Umbra Et Imago	
German Hip Hop		
Absolute Beginner	Blumentopf	Die Fantastischen Vier
Fettes Brot	Kinderzimmer Productions	Thomas D
Hard Core Rap		
50 Cent	Busta Rhymes	Dr. Dre
Genius	Gravediggaz	Onyx
Heavy Metal / Thrash		
Crowbar	Megadeth	Metallica
Pantera	Sepultura	
Italian		
Adriano Celentano	Angelo Branduardi	Eros Ramazzotti
Tiziano Ferro	Zucchero	

Table A-4: Composition of collection *C103a22g*.

Jazz		
Dave Brubeck	Ernest Ranglin	George Benson
Jazzkantine		
Jazz Guitar		
Barney Kessel	Herb Ellis & Joe Pass	Kenny Burrell
Martin Taylor	Tuck Andress	
Melodic Metal		
Evanescence	Heavenly	Lacuna Coil
Nightwish	Stratovarius	
Punk		
Bad Religion	Blink 182	Die Goldenen Zitronen
Green Day	Offspring	Rancid
Reggae		
Bob Marley & The Wailers	Capleton	Eek A Mouse
Trance		
Cosmic Gate	Darude	Gigi d'Agostino
Scooter	The Speed Freak	
Trance2		
Hallucinogen	Koxbox	Manmademan
Ominus		

Table A-5: Continuation of Table A-4.

A Cappella		
Golden Gate Quartet	Ladysmith Black Mambazo	The Cafe of the Gate of Salvation
The Heavenly Light Quartet		
Acid Jazz		
Count Basic	Jazzanova	Saint Germain
Us3		
Blues		
Etta James	John Lee Hooker	Muddy Waters
Willie Dixon		
Bossa Nova		
Antonio Carlos Jobim	Baden Powell	João Gilberto
Stan Getz, João & Astrud Gilberto		
Celtic		
Clannad	Enya	Lunasa
Sharon Shannon	The Chieftains	
Electronic		
2 Unlimited	Ace Of Base	Aphex Twin
Cosmic Gate	Culture Beat	Darude
Dillinja	DJ Shadow	DJ Zinc
Ed Rush & Optical	Gigi d'Agostino	Goldie
Grooverider	Hallucinogen	Kaito
Ken Ishii	Koxbox	Kruder & Dorfmeister
Magic Affair	Manmademan	Massive Attack
Masterboy	Matthew Herbert	Ominus
Scooter	Sofa Surfers	The Speed Freak
U96		
Folk Rock		
Corvus Corax	In Extremo	Schandmaul
Subway To Sally	Umbra Et Imago	
Italian		
Adriano Celentano	Angelo Branduardi	Eros Ramazzotti
Tiziano Ferro	Zucchero	
Jazz		
Barney Kessel	Dave Brubeck	Ernest Ranglin
George Benson	Herb Ellis & Joe Pass	Jazzkantine
Kenny Burrell	Martin Taylor	Tuck Andress
Metal		
Borknagar	Cannibal Corpse	Crowbar
Dimmu Borgir	Entombed	Evanescence
Heavenly	Lacuna Coil	Megadeth
Metallica	Nightwish	Pantera
Sepultura	Stratovarius	

Table A-6: Composition of collection *C103a13g*.

Punk Rock		
Bad Religion	Blink 182	Die Goldenen Zitronen
Green Day	Offspring	Rancid
Rap		
50 Cent	Absolute Beginner	Blumentopf
Busta Rhymes	Die Fantastischen Vier	Dr. Dre
Fettes Brot	Genius	Gravediggaz
Kinderzimmer Productions	Onyx	Thomas D
Reggae		
Bob Marley & The Wailers	Capleton	Eek A Mouse

Table A-7: Continuation of Table A-6.

	absolute				relative			
	AMG tier				AMG tier			
Genre	1	2	3	Σ	1	2	3	Σ
Blues	37	95	56	**188**	0.20	0.51	0.30	**9.4%**
Electronica	25	68	2	**95**	0.26	0.72	0.02	**4.8%**
Reggae	28	32	0	**60**	0.47	0.53	0.00	**3.0%**
Jazz	93	400	318	**811**	0.11	0.49	0.39	**40.7%**
Folk	44	36	1	**81**	0.54	0.44	0.01	**4.1%**
Heavy Metal	14	59	198	**271**	0.05	0.22	0.73	**13.6%**
RnB	47	82	73	**202**	0.23	0.41	0.36	**10.1%**
Country	39	132	75	**246**	0.16	0.54	0.30	**12.3%**
Rap	33	8	0	**41**	0.80	0.20	0.00	**2.1%**
Total	360	912	723	**1995**	0.18	0.46	0.36	**100.0%**

Table A-8: Distribution of genres and tiers given by AMG in collection *C1995a9g*.

Angra	Annihilator	Anthrax
Apocalyptica	Bad Religion	Black Sabbath
Blind Guardian	Borknagar	Cannibal Corpse
Century	Crematory	Deicide
Dimmu Borgir	Edguy	Entombed
Evanescence	Finntroll	Gamma Ray
Green Day	Guano Apes	Hammerfall
Heavenly	HIM	Iron Maiden
Iron Savior	Judas Priest	Krokus
Lacuna Coil	Lordi	Majesty
Manowar	Metal Church	Metallica
Motörhead	Nightwish	Nirvana
Offspring	Pantera	Paradise Lost
Pink Cream 69	Powergod	Primal Fear
Rage	Regicide	Scorpions
Sepultura	Soulfly	Stratovarius
Tiamat	Type O Negative	Within Temptation

Table A-9: List of artists in collections *C51a240m* and *C51a499m*.

Artist	Album
2 Unlimited	The Real Thing
AC-DC	Ballbreaker
Alice Cooper	Brutal Planet
Alice Cooper	Classicks
Angelo Branduardi	La Pulce d'Acqua
Angra	Angels Cry
Angra	Fireworks
Angra	Rebirth
Annihilator	Carnival Diablos
Anthrax	Armed And Dangerous
Apollo 440	Electro Glide In Blue
Ärzte	13
Ärzte	Die Bestie in Menschengestalt
Ärzte	Planet Punk
Ash	1977
Ayreon	The Dream Sequencer
Babylon Zoo	Spaceman
Bad Religion	No Substance
Bad Religion	Stranger Than Fiction
Bad Religion	The Gray Race
Bad Religion	The New America
Bad Religion	The Process Of Belief
Beautiful World	In Existence
Billy Joel	River Of Dreams
Black Sabbath	Forbidden
Black Sabbath	Paranoid
Blind Guardian	Nightfall In Middle-Earth
Blind Petition	The Elements Of Rock
Bloodhoundgang	Hooray For Boobies
Blue Öyster Cult	Cult Classic
Blue Öyster Cult	Heaven Forbid
Bryan Adams	So Far So Good
Century	Melancholia
Century	The Secret Inside
Clawfinger	Use Your Brain
Crematory	Act Seven
Crematory	Believe
Crematory	Early Years
Culture Beat	Got To Get It
Culture Beat	Serenity
Deep Purple	Purplexed
Dexy's Midnight Runners	Too-Rye-Ay
Dimmu Borgir	Godless Savage Garden
Dire Straits	On The Night
East 17	It's Alright

Table A-10: Artist and album names in collection *C255b*.

Artist	Album
EAV	Geld Oder Leben
EAV	Himbeerland
EAV	Im Himmel Ist Die Hölle Los
EAV	Let's Hop
EAV	Nie Wieder Kunst
EAV	Watumba
Edguy	The Savage Poetry
Edguy	Theater Of Salvation
Edguy	Vain Glory Opera
Enya	Paint The Sky With Stars
Enya	The Memory Of Trees
Eric Clapton	Reptile
Faith No More	King For A Day
Falco	The Final Curtain
Filter	Short Bus
Fish	Raingods With Zippos
Foo Fighters	There Is Nothing Left To Lose
France Gall	Greatest Hits
Freddie Mercury	Living On My Own
Gamma Ray	Blast From The Past
Gary Moore	Back To The Blues
Gary Moore	Corridors Of Power
Gary Moore	Dirty Fingers
Gary Moore	Run For Cover
General Base	Base Of Love
Goldenen Zitronen	Punkrock
Green Day	Dookie
Green Day	Insomniac
Green Day	Nimrod
Green Day	Warning
Herbert Grönemeyer	Bleibt Alles Anders
Guano Apes	Don't Give Me Names
Guano Apes	Proud Like A God
H.I.M.	Greatest Love Songs Vol. 666
H.I.M.	Razorblade Romance
Haddaway	The Album
HammerFall	Crimson Thunder
HammerFall	Legacy Of Kings
HammerFall	Renegade
H-Blockx	Discover My Soul
Helloween	Metal Jukebox
Ice MC	Take Away The Colour
Imperio	Quo Vadis
In Extremo	Die Verrückten Sind In Der Stadt
In Extremo	Verehrt und Angespien

Table A-11: Continuation of Table A-10.

Artist	Album
In Extremo	Weckt die Toten!
Iron Maiden	Can I Play With Madness
Iron Maiden	Piece Of Mind
Iron Maiden	Seventh Son Of A Seventh Son
Iron Maiden	Virtual XI
J.B.O.	Blastphemie
J.B.O.	Live Sex
Jam Tronik	I'd Do Anything for Love
Jean Michel Jarre	Oxygene 7-13
Jazz Gitti	Der Bauch Mu Weg
Joan Jett	The Original Hit Collection
Kansas	Point Of Know Return
Lacuna Coil	In A Reverie
Led Zeppelin	Remasters
Let Loose	Crazy For You
Lunasa	Otherworld
Lunasa	The Merry Sisters Of Fate
Magic Affair	Night Of The Raven
Magic Affair	Omen III
Manowar	Hell On Wheels Live
Manowar	Louder Than Hell
Manowar	The Hell Of Steel
Manowar	The Triumph Of Steel
Marillion	Fugazi
Mark 'Oh	Love Song
Mark 'Oh	Never Stop That Feeling
Masterboy	Feel The Heat Of The Night
Meat Loaf	Bat Out Of Hell II
Metal Church	Hanging In The Balance
Metal Church	Live
Metallica	Bay Area Trashers
Metallica	Black Album
Metallica	Garage Inc.
Metallica	Kill'em All
Metallica	Load
Metallica	Reload
Metallica	Ride The Lightning
Metallica	S&M
Mike Oldfield	Guitars
Mike Oldfield	Tubular Bells III
Nana	Nana
Neil Young	Sleeps With Angels
Neil Young	Weld
Nightwish	Over The Hills And Far Away
Nirvana	In Utero

Table A-12: Continuation of Table A-11.

Artist	Album
Nirvana	Incesticide
Nirvana	Nevermind
Offspring	Americana
Offspring	Conspiracy Of One
Offspring	Ignition
Offspring	Ixnay On The Hombre
Offspring	Smash
Offspring	The Offspring
Paradise Lost	Draconian Times
Paradise Lost	Host
Paradise Lost	One Second
Paradise Now!	Erica
Pearl Jam	Live In Atlanta
Pearl Jam	No Code
Pearl Jam	Pearl Jam
Pearl Jam	Vitalogy
Pink Cream 69	Electrified
Pink Cream 69	Endangered
Powergod	Evilution Part I
Primal Fear	Jaws Of Death
Primal Fear	Nuclear Fire
Prodigy	No Good
Queen	A Day At The Races
Queen	A Kind Of Magic
Queen	A Night At The Opera
Queen	Flash Gordon
Queen	Greatest Hits
Queen	Greatest Hits II
Queen	Hot Space
Queen	Innuendo
Queen	Jazz
Queen	Live At Wembley '86
Queen	Live Killers
Queen	Live Magic
Queen	Made In Heaven
Queen	News Of The World
Queen	Queen I
Queen	Queen II
Queen	Rocks
Queen	The Game
Queen	The Miracle
Queen	The Works
Queensrche	Promised Land
R.E.M.	Monster
R.E.M.	Murmur

Table A-13: Continuation of Table A-12.

Artist	Album
Rammstein	Du Hast
Rammstein	Mutter
Rancid	Life Won't Wait
Schandmaul	Narrenkönig
Peter Schilling	Major Tom
Scooter	How Much Is The Fish?
Scooter	Move Your Ass!
Scorpions	Animal Magnetism
Scorpions	Best Of Scorpions Vol. 2
Scorpions	Eye II Eye
Scorpions	Face The Heat
Scorpions	Live Bites
Scorpions	Pure Instinct
Scrub	Wake Up!
Sisters Of Mercy	Some Girls Wander By Mistake
Ska-P	El Vals Del Obrero
Sorry About Your Daughter	Face
Steel Prophet	Messiah
Stiltskin	Inside
Stratovarius	Destiny
Stratovarius	Infinite
Subway To Sally	Herzblut
Tangerine Dream	The Park Is Mine
Therapy?	Pleasure Death
To Die For	All Eternity
Tom Petty	Echo
Toten Hosen	Auswärtsspiel
Toten Hosen	Bis Zum Bitteren Ende
Toten Hosen	Im Auftrag des Herrn
Toten Hosen	Learning English
Toten Hosen	Opium fürs Volk
Toten Hosen	Reich & Sexy
Toten Hosen	Unsterblich
Troggs	Rock Classics 4
Type O Negative	The Least Worst Of
U 96	Heaven
U 96	Inside Your Dreams
Van Halen	1984
Van Halen	Van Halen 3
Westernhagen	So Weit
Westernhagen	Westernhagen
White Lion	The Best Of White Lion
Xysma	De Luxe
ZZ Top	Greatest Hits
ZZ Top	XXX

Table A-14: Continuation of Table A-13.

Appendix B: Artist Similarity Evaluation on the AGMIS Collection

To assess the performance of the TF·IDF approach presented in Subsection 3.3.3 on a real world collection, the author conducted k-NN classification experiments using the AGMIS collection of 636,475 artists. To estimate how well the given genres are distinguished by the similarities calculated on TF·IDF vectors, an artist-to-genre classification setting was used. The k-NN classification experiments were conducted using leave-one-out cross-validation. In case that the most frequent genre in the set of nearest neighbors was ambiguous[1], a randomly chosen genre out of those with a maximum number of occurrences in the k-NN set was predicted.

Table B-1 summarizes the results. The table gives, for each genre, its share in the collection (column *Proportion*) and the classification accuracies for different k-NN evaluations. Furthermore, the last row shows the weighted average accuracies. The proportion of each genre in the collection can be regarded as a baseline since a naive classifier predicting a genre with a probability that equals the genre's share in the collection would approximate these results in the long run. From Table B-1 it can be seen that, even though the results are not outstanding, they consistently exceed the baseline. Taking the different degrees of coherency of the genres into consideration, it seems reasonable to conclude that more specific and coherent genres, like "Celtic" or "Classical", perform better (in relation to their small share in the collection) than genres spanning a wide range of different styles of music, like "Rock" or "Electronica". As the AGMIS collection is based on the genre taxonomy by AMG, in particular, the genre "Rock" encompasses quite inhomogeneous music, such as synthesizer pop and death metal.

Analyzing the changes in accuracy when considering different values of k, two effects have to be taken into account. On the one hand, the larger the share of a genre in the collection, the higher the likelihood that artists from this genre may occur coincidentally in the set of nearest neighbors when increasing the value of k. Thus, for very large values of k, the genre "Rock", which is by far the most frequent one, would be predominantly predicted. This trend is easily discernible from the positive correlation between accuracy and k value for this genre. On the other hand, it is interesting that some genres show accuracy values that are largely independent of growing k values, e.g., "Blues", "Jazz", "Reggae", and "World", or whose accuracy values increase, although the share of the respective genre in the collection is rather small. For example, even though the amount of classical artists in the collection is less than 2

[1] For example, in a 5-NN experiment, two out of the nearest neighbors may be rock artists, another two may be blues artists, and one may be a rap artist.

percent, the accuracy of this genre rises by more than 10 percentage points when comparing the 1-NN to the 20-NN experiment. The genre "Celtic", with a share of less than 1 percent in the collection, performs similarly, even though the increase in accuracy is smaller in this case.

Genre	Proportion	1-NN	3-NN	5-NN	10-NN	20-NN
Avantgarde	0.70 %	12.47	10.01	9.18	8.26	6.17
Blues	2.14 %	17.86	18.06	17.97	17.03	16.11
Celtic	0.61 %	26.72	27.71	29.42	30.00	29.12
Classical	1.77 %	29.09	32.23	35.73	38.20	39.45
Country	2.56 %	20.58	21.11	21.65	21.18	19.70
Easy Listening	0.78 %	8.83	6.75	5.39	3.71	2.05
Electronica	5.54 %	23.92	25.13	25.42	24.51	23.51
Folk	2.16 %	11.50	9.57	8.25	6.48	4.94
Gospel	4.15 %	23.35	23.90	23.68	23.16	21.69
Jazz	10.00 %	29.68	31.08	32.43	32.13	30.43
Latin	5.31 %	33.00	34.32	35.68	36.52	35.45
New Age	2.10 %	12.45	11.41	10.66	9.44	8.21
Rap	4.14 %	23.43	24.13	24.20	23.07	20.95
Reggae	1.34 %	24.51	25.56	26.09	25.83	25.39
RnB	3.39 %	11.19	9.62	8.53	6.89	5.06
Rock	42.08 %	51.77	60.34	71.36	81.30	87.30
Vocal	1.84 %	10.43	9.68	9.09	7.52	6.19
World	9.39 %	32.19	32.65	33.90	33.46	31.61
Total	**100.00 %**	**35.94**	**39.85**	**44.81**	**48.69**	**50.39**

Table B-1: Accuracies, in percent, for k-NN evaluations using the TF·IDF approach on the AGMIS collection.

Biography

Markus Schedl graduated in Computer Science at the *Vienna University of Technology* in March 2004. Since 2003 he has been studying International Business Administration at the *Vienna University of Economics and Business Administration*. In October 2004 he started working on his doctoral thesis in Computer Science under the supervision of Prof. Dr. Gerhard Widmer at the Department of Computational Perception at the *Johannes Kepler University Linz*. While working on his PhD thesis, he (co-)authored more than 25 refereed conference papers and 3 journal articles. Furthermore, Markus Schedl reviewed submissions for the *International Conference on Music Information Retrieval (ISMIR)*, the *European Conference on Artificial Intelligence (ECAI)*, the *IEEE International Conference on Digital Information Management (ICDIM)*, the *IEEE Visualization Conference (Vis)*, and the *IEEE Information Visualization Conference (InfoVis)*. He also reviewed articles for the journals *IEEE Transactions on Multimedia* and *Springer Multimedia Systems*, as well as for the *IEEE Communications Magazine*. His main research interests include Web Mining, Music and Multimedia Information Retrieval, Information Visualization, Intelligent User Interfaces, and Financial Markets.