On the Use of the Web and Social Media in Multimodal Music Information Retrieval

Narrowing the Gap between Systems and Users

Habilitationsschrift

Dipl.-Ing. Mag. Dr.
Markus Schedl

JKU
JOHANNES KEPLER
UNIVERSITÄT LINZ

Habilitationsschrift von Markus Schedl May 2013

For my wonderful daughter, Alina Laura.

"Aerodynamically, the bumble bee shouldn't be able to fly,
but the bumble bee doesn't know it so it goes on flying anyway."

— Mary Kay Ash

"The most exciting phrase to hear in science, the one that
heralds new discoveries, is not 'Eureka!' but 'That's funny...'"

— Isaac Asimov

Contents

Biography

Markus Schedl graduated in Computer Science from the *Vienna University of Technology*. He earned his Ph.D. degree in Computational Perception from the *Johannes Kepler University Linz*, where he is employed as Assistant Professor at the Department of Computational Perception. He further studied International Business Administration at the *Vienna University of Economics and Business Administration* as well as at the *Handelshögskolan of the University of Gothenburg*, which led to a Master's degree.

Markus (co-)authored more than 70 refereed conference papers and journal articles (among others, published in ACM Multimedia, SIGIR, ECIR, IEEE Visualization; Journal of Machine Learning Research, ACM Transactions on Information Systems, Springer Information Retrieval, IEEE Multimedia). Furthermore, he serves on various program committees and reviewed submissions to several conferences and journals (among others, ACM Multimedia, ECIR, IJCAI, ICASSP, IEEE Visualization; IEEE Transactions of Multimedia, Elsevier Data & Knowledge Engineering, ACM Transactions on Intelligent Systems and Technology, Springer Multimedia Systems). His main research interests include web and social media mining, information retrieval, multimedia, music information research, and personalized user interfaces. He is co-founder of the *International Workshop on Advances in Music Information Research* (AdMIRe) and co-organizer of the 3rd *International Workshop on Search and Mining User-generated Contents* (SMUC).

Markus leads the FWF stand-alone projects *Personalized Music Retrieval via Music Content, Music Context, and User Context* (P22856) and *Social Media Mining for Multimodal Music Retrieval* (P25655). Furthermore, he is work package leader in the FP7-ICT-2011-9 STREP project *PHENICX — Performances as Highly Enriched aNd Interactive Concert eXperiences* (601166).

Since 2007 Markus has given several lectures, for instance, *Music Information Retrieval*, *Exploratory Data Analysis*, *Multimedia Search and Retrieval* and *Learning from User-generated Data*, the latter two being in preparation. He further spent several guest lecturing stays at the *Universitat Pompeu Fabra, Barcelona, Spain*, the *Utrecht University, the Netherlands*, the *Queen Mary, University of London, UK*, and the *Kungliga Tekniska Högskolan, Stockholm, Sweden*.

Acknowledgments

First and foremost, I would like to thank my boss and mentor, *Gerhard Widmer*. In particular, I highly appreciate his outstanding scientific excellence, his great support, his modesty, and his efforts to establish a very stimulating working environment. I would like to thank Gerhard especially for giving me the freedom to perform interesting research, according to my own interests. Despite this self-determined and productive research environment, Gerhard always cared and offered support when needed.

I further take the opportunity to thank my colleagues for their support, both in scientific and personal matters. Special thanks go to *Peter Knees*, my long-lasting room mate, for many interesting discussions and collaborations; to *David Hauger* for implementing my sometimes crazy ideas; to *Sebastian Böck* and *Maarten Grachten* for their countless helpful suggestions on shell scripting and Python programming; and to *Claudia Kindermann* for her administrative support.

Furthermore, I wish to express my gratitude to all the great people with whom I had the pleasure to collaborate during the past few years. Not being able to exhaustively list everyone here (please apologize), I particularly want to highlight the collaborations with *Dominik Schnitzer*, *Arthur Flexer*, *Jan Schlüter*, and *Marcin Skowron* at the Austrian Research Institute for Artificial Intelligence, Vienna; with *Julián Urbano* at the Universidad Carlos III de Madrid, Spain; with *Francesco Ricci* and *Marius Kaminskas* at the Free University of Bozen–Bolzano, Italy; with *Bogdan Ionescu* at the University Politehnica of Bucharest, Romania; with *Òscar Celma* at Gracenote, Inc., USA; and with *Katayoun Farrahi* of the Department for Pervasive Computing. For our joint effort in establishing the `MusiClef` evaluation forum, I wish to thank *Cynthia Liem* (Delft University of Technology, the Netherlands), *Nicola Orio* (University of Padova, Italy), and *Geoffroy Peeters* (Institut de Recherche et Coordination Acoustique/Musique, Paris, France).

For providing the opportunity to spend guest lectures at their respective universities, I would like to thank *Frans Wiering*, *Ad Feelders*, and *Remco Veltkamp* (Universiteit Utrecht, the Netherlands); *Emilia Gómez* and *Xavier Serra* (Universitat Pompeu Fabra, Barcelona, Spain); *Simon Dixon* and *Mark Plumbley* (Queen Mary University, London, UK); *Hedvig Kjellström* and *Anders Friberg* (Kungliga Tekniska Högskolan, Stockholm, Sweden). For her administrative support in preparing these guest lectures, I highly appreciate the work of *Petra Lehner*.

Last but not least, I would like to thank all committee members of my postdoctoral lecture qualification board and all reviewers of this thesis for their highly valuable feedback.

My special thanks go to *Cornelia* and *Alina Laura* for their love and indulgence.

Abstract

This postdoctoral thesis elaborates on the exploitation of multiple data sources to build *multimodal music access systems*. Multimodality refers to two conceptual aspects: (i) different modalities in *modeling human music perception*, such as the use of different multimedia material to infer perceptual features, and (ii) different modalities involved in the process of *accessing music*. Both aspects are reflected in the selection of papers constituting this thesis. The former aspect can be detailed further into factors describing the *music content*, the *music context*, *user properties*, and the *user context*. The music content relates to information extracted from the audio signal; the music context comprises factors not encoded in the audio, nevertheless important to human music perception; user properties refer to static characteristics of the listener, while the user context is represented by dynamic factors of the user, both intrinsic and environmental.

This thesis particularly researches web and social media sources to extract multimodal information, including the following:

- similarities between music items (for instance, artists and songs),
- descriptive labels (also known as "tags"),
- "prototypicality" of artists for a genre,
- images of album cover artwork,
- members and instrumentation of bands,
- country of origin of artists or bands,
- popularity of artists or bands (at the country level),
- spatiotemporal music listening activities of users, and
- "mainstreaminess" of a population's music taste.

Based on these different pieces of information, intelligent music access systems (such as music retrieval, recommendation, or browsing systems) are elaborated. Although the main focus of this thesis is inferring *music context* and *user context* information from the web and social media, a prototypical music browsing interface that combines music content and contextual information is also included.

The following page contains a list of publications which were selected by the author to constitute the core part of this thesis. The main contributions of the thesis, including references to the respective publications, can be summarized as follows:

- **a critical investigation of the largely neglected role of the user in current research on Music Information Retrieval [A],**
- **novel techniques to derive semantic information from the web and social media, related to music items [B,C,E,F], music listening activity of users [D,H], and music preferences of entire populations [C,D],**
- **comprehensive investigations of models to infer music similarity from the web and from social media [E,F] and to combine these context-based similarities with audio-based ones and with user context aspects, and**
- **a system to explore music collections, which combines music content and contextual data [G].**

Institut für Computational Perception
Johannes Kepler Universität, Linz, Austria

[A] Markus Schedl, Arthur Flexer, and Julián Urbano.

The Neglected User in Music Information Retrieval Research. *Journal of Intelligent Information Systems*, 2013.

[B] Markus Schedl, Gerhard Widmer, Peter Knees, and Tim Pohle.

A Music Information System Automatically Generated via Web Content Mining Techniques. *Information Processing & Management*, 47, 2011.

[C] Markus Schedl, Tim Pohle, Noam Koenigstein, and Peter Knees.

What's Hot? Estimating Country-Specific Artist Popularity. In *Proceedings of the 11th International Society for Music Information Retrieval Conference* (ISMIR), Utrecht, the Netherlands, August 2010.

[D] Markus Schedl.

Leveraging Microblogs for Spatiotemporal Music Information Retrieval. In *Proceedings of the 35th European Conference on Information Retrieval* (ECIR), Moscow, Russia, March 2013.

[E] Markus Schedl, Tim Pohle, Peter Knees, and Gerhard Widmer.

Exploring the Music Similarity Space on the Web. *ACM Transactions on Information Systems*, 29(3), July 2011.

[F] Markus Schedl.

#nowplaying Madonna: A Large-Scale Evaluation on Estimating Similarities Between Music Artists and Between Movies from Microblogs. *Information Retrieval*, 15:183–217, June 2012.

[G] Markus Schedl, Christian Höglinger, and Peter Knees.

Large-Scale Music Exploration in Hierarchically Organized Landscapes Using Prototypicality Information. In *Proceedings of the ACM International Conference on Multimedia Retrieval (ICMR)*, Trento, Italy, April 2011.

[H] Markus Schedl, Dominik Schnitzer.

Hybrid Retrieval Approaches to Geospatial Music Recommendation. In *Proceedings of the 35th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, Dublin, Ireland, July–August 2013.

Part I

Introduction and Contributions

Institut für Computational Perception
Johannes Kepler Universität, Linz, Austria

1 Introduction to Music Information Retrieval

Music is omnipresent in our daily lives and is a valuable treasure of mankind. Since almost everyone enjoys listening to music, and accordingly has an opinion about different styles of music, research that analyzes, processes, represents, and eventually provides access to music is a highly important field. Addressing these tasks, the area of Music Information Retrieval (MIR), despite being a relatively young field, has attracted quite a lot of attention since its emergence in the late 1990s. As MIR is a highly multidisciplinary field, it took a while until first definitions widely agreed on emerged:

J. Stephen Downie in [15]:

> MIR is a *multidisciplinary* research endeavor that strives to develop innovative *content-based searching schemes*, *novel interfaces*, and *evolving networked delivery mechanisms* in an effort to make the world's vast store of music accessible to all.

Markus Schedl in [60]:

> 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 first definition highlights the importance of content-based retrieval, which in the field of MIR comprises extracting audio signal-based features and using them (or derived, higher level information) for music representation or access, including but not restricted to the narrow retrieval paradigm typically employed in Information Retrieval (IR). According to this paradigm, (i) a query string is taken as input, (ii) is converted into a weighted term vector representation, (iii) is compared to similar representations of documents in a corpus, and (iv) a set of documents most relevant to the query is returned.
The second definition focuses on the aspects of extracting and using information about music items that are given in various representation flavors, not limited to the actual audio. This presumably multimodal information can be used again in various types of music access schemes, such as retrieval, browsing, or query-by-humming.

The use of *social media*, in particular of microblogging services, has been spiraling during the past couple of years. According to latest official figures as of April 2011, today's most popular microblogging service, `Twitter`[1], has more than 200 million registered users[2] who are creating a billion posts every week[3]. Newer, but unofficial sources report 175 million tweets posted every day throughout the year 2012[4], the existence of about 530 million `Twitter` accounts as of July 2012, and a total of 163 billion tweets since the start of the microblogging service[5].

Judging from these sheer numbers, harvesting and analyzing huge amounts of microblogs is an extremely challenging task. However, the abundance of user-generated data posted on microblogging platforms is only one reason for the difficulty of the problem. Another one is the high amount of noise inherent to almost all kinds of user-generated data. For instance, a

[1] `http://www.twitter.com`
[2] `http://huffingtonpost.com/2011/04/28/twitter-number-of-users_n_855177.html`
[3] `http://blog.twitter.com/2011/03/numbers.html`
[4] `http://www.mediabistro.com/alltwitter/twitter-stats_b32050`
[5] `http://diegobasch.com/some-fresh-twitter-stats-as-of-july-2012`

lot of spam is found in microblogs and people often just tweet irrelevant or even pointless things. Making out information relevant to the task at hand therefore resembles finding a needle in a haystack.

The field of Social Media Mining (SMM) faces these challenges, for instance, by developing highly efficient processing and analysis techniques for text and other kinds of media typically found on social media platforms. Inferring and making use of semantic information from these multimedia data sources, with a particular focus on information about music, constitutes the major part of this thesis. Combined with information derived from the music content, i.e. the audio, exciting applications that might revolutionize music access are not unlikely to emerge in the near future.

The remainder of this thesis is organized as follows. A discussion of multimodal aspects in MIR research is provided in Chapter 2, particularly focusing on perceptual aspects and access schemes. In Chapter 3, a literature overview illustrating the state-of-the-art in related areas is presented and open challenges are identified. Chapter 4 details how these challenges are faced by the publications selected for this thesis. The chapter further summarizes the main scientific contributions of this thesis and provides links to the papers constituting the core part of the thesis, which follows thereafter.

2 Multimodal Music Information Retrieval

Multimodality in the context of MIR may refer to at least two conceptual aspects: (i) integrating different modalities in *models of music perception* (such as different multimedia material) [38] and (ii) using different modalities while *accessing music collections* [69]. Please note that these two aspects may overlap in some approaches or applications. For instance, rhythmic information may be incorporated into a music similarity model, but tapping a rhythm may also serve as query to a music retrieval system. Both interpretations of multimodality are discussed in the following and will be addressed in this thesis.

2.1 Multimodality in Perceptual Aspects

During a scientific seminar on "Multimodal Music Processing", which took place in January 2011 in the Schloss Dagstuhl, Leibnitz-Zentrum für Informatik, Wadern, Germany, the author led a discussion group about the role of the user in MIR. We discussed different aspects of user-centric MIR, and highlighted the (until recently) largely neglected role of the user in MIR research. These discussions eventually resulted in a categorization scheme of aspects influencing human music perception, an extension of the one suggested in [85]. As Figure 1 shows, the categorization entails severals multimodal factors.

Music Content

Factors categorized as *music content* are defined as human perceptual aspects that can be extracted from the audio signal with state-of-the-art methods. The corresponding music features range from low-level representations such as *Mel Frequency Cepstral Coefficients* (MFCC) [1, 39] to mid-level features such as *attackness* [55, 45] or *harmony* [19]. High-level features, typically defined in a fuzzy way as "understandable by everyone", are commonly semantic music descriptors, for instance, collaborative tags. Although it is infeasible to automatically extract such labels from the raw audio signal without any additional information, using state-of-the-art music auto-tagging approaches such as [91, 92] enables learning and

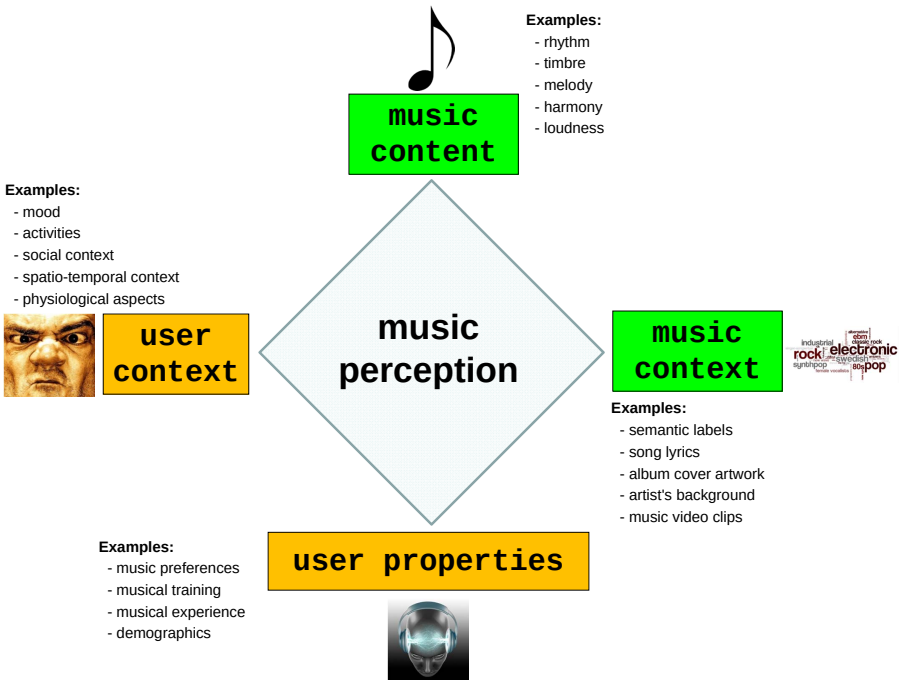


Figure 1 Categorization of factors that influence human music perception.

predicting relations between acoustic features and semantic labels. More details on music content-based MIR can be found in [9].

Music Context

Music context refers to all aspects that are not encoded in the audio signal, nevertheless influence our perception of music. These aspects are typically multimodal and represented as high-level features. They span a wide range, from information about the political background of a songwriter to album cover artwork to the semantics of song lyrics to user-generated music video clips. Most of the aspects addressed in the publications constituting this thesis belong to this category (e.g., similarity and popularity estimates from microblogs, band membership information from web pages). The corresponding computational features are foremost textual (e.g., semantic labels inferred from music-related web pages) or numerical (e.g., estimates of artist popularity). A more detailed elaboration on contextual factors in MIR can be found in two book chapters written by the author of this thesis [63, 65].

User Properties

User properties refer to constant or slowly changing characteristics of the user, such as her music taste, education or skills in playing instruments. Also demographic information belong to this category. In the context of this thesis, such properties are reflected in listening histories (e.g., derived from microblogs or dedicated social media platforms like `Last.fm`[6]). They can

[6] `http://www.last.fm`

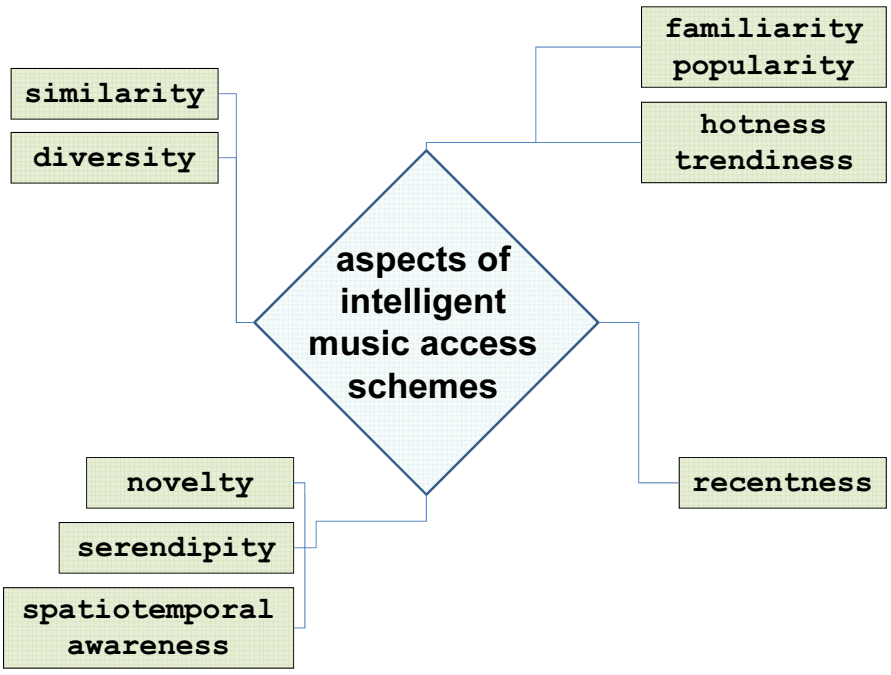


Figure 2 Multimodal aspects to consider in intelligent music access models.

be used to model an overall user profile, which is adapted according to user context factors (see below). Such profiles play a vital role in elaborating personalized and context-aware music) retrieval systems.

User Context

User context aspects represent dynamic, frequently changing factors, such as the user's current activity, social context, or environmental aspects like location, time, or weather conditions. Such aspects are addressed in this thesis via spatial and temporal information about listening events, inferred from microblogs, assuming that they are highly dynamic. For location information, the dynamic obviously depends on the point of view; from a global perspective, a person who spends almost all of her time in the same city does not show high dynamics in the location dimension, while from a local perspective, we can distinguish between different locations (e.g., home, work place, shopping center). However, such a precise localization is usually neither possible, nor desired by the user, due to privacy concerns.

2.2 Multimodality in Access Scheme

The author of this thesis proposes in [69] a model that includes several aspects to consider when elaborating intelligent music access schemes. These aspects go far beyond those employed in traditional metadata-based search facilities still common in music retrieval systems, i.e., using as query strings like artist name, song name, or music style. Figure 2 displays the factors which are deemed vital when building such intelligent systems.

Similarity

Similarity is a vital concept in the fields of Information Retrieval (IR) and Recommendation Systems (RS). Similarity measures typically quantify the resemblance of queries and documents in IR and of items and/or users is RS. In MIR, similarity measures are commonly defined between two music items, such as songs or artists.

In a multimodal music access system, different aspects of similarity should be taken into account. Those correspond to aspects of music content and music context, as defined above. Assuming that our perception of musical similarity is affected by user-centric factors as well, user properties and user context should be integrated too. As discussed in [67], it is reasonable to assume such a subjective notion of similarity. To give an example, a fan of Heavy Metal music might perceive a Viking Metal song as highly dissimilar from a Melodic Metal piece, while for the majority of people the two will sound alike.

In the publications selected for this thesis, similarity aspects are either derived from web pages [80] [E], from microblogs [64] [F], from the audio signal [70] [G], or from a combination of these [83] [H]. When talking about textual data sources, it has to be noted that short text snippets such as microblogs require special processing when used for similarity and retrieval tasks [94]. Reported in [64], we accordingly identify different requirements when modeling similarity estimators from web pages, compared to modeling them from microblogs.

Diversity

Even though the output of a music retrieval system should obviously contain music items similar to the input query or seed, the results should also exhibit a certain degree of diversity. This is of particular importance for tasks such as automated playlist generation or music recommendation, because otherwise the system is likely to leave the user bored by suggesting over and over again music that all sounds the same. Indeed, in a user study conducted to assess our approach to audio-based automated playlist generation [54], we found that users often prefer playlists with slightly higher stylistic entropy over playlists in which the music items are arranged by minimizing acoustic distances between consecutive tracks. Producing a well-diversified result set for a given query is thus a common requirement for IR systems [11].

In the context of MIR, the so-called "album effect" [102] should be mentioned. This effect relates to the fact that, due to the same recording environment and parameters, tracks on the same album usually show a higher level of audio similarity than other tracks (even by the same artist). To alleviate this issue, some music retrieval systems omit in their result set tracks from the same album or by the same artist as the seed.

Familiarity/Popularity

Familiarity or popularity measures how well-known a music item or an artist is. Popularity has a more positive connotation than the neutral expression of familiarity. However, we will use the terms interchangeably in the remainder. Please note that popularity can hardly be defined without considering the spatial and temporal context. The temporal context distinguishes familiarity from hotness (see below), whereas the spatial context plays a vital role in modeling individual as well as group user profiles. According to the temporal dimension, popularity can be regarded as a longer lasting property, whereas hotness usually relates to recent appreciation of typically shorter duration, although hot artists might also be very familiar to many people. To give an example, "The Beatles" are certainly popular, whereas "Lady Gaga" currently tends to rank higher on the hotness dimension, as of the time of writing. As for the spatial dimension, for instance, Italians are very familiar with the

artist "Tiziano Ferro", whereas most US-Americans may never have heard of him. Another example is "Nithyasree Mahadevan", who is an eminent Carnatic musician, well-known in India, but unknown almost everywhere esle in the world.

Hotness/Trendiness

In contrast to familiarity or popularity, the aspect of hotness or trendiness (used interchangeably in the following) relates to the amount of buzz or attention an artist, album, or song is currently attracting[7]. While popularity refers to the overall familiarity of a population with a music item, hotness describes the popularity in the recent past or at the moment. A current example of a trendy song as of the time of writing is "Gangnam Style" by "Psy".

Recentness

The measurement of recentness distinguishes recently released songs or albums from items that are older and hence have a longer history. Recentness is thus related to hotness in terms of temporal closeness to the present, although recent music items do not necessarily have to originate from hot artists, of course.

Novelty

Music recommendation systems that keep on suggesting tracks or artists known to the user are unlikely to satisfy his or her information and entertainment needs, even if the recommendations perfectly match the user's musical taste. Providing unknown (and interesting) recommendations to the user is hence a vital requirement for an intelligent music retrieval system.

Serendipity

Serendipity refers to the fact that a user is surprised in a positive way since she discovered an item she did not expect or was not aware of. A recommendation or retrieval system that is able to make serendipitous suggestions is hence highly beneficial for increasing user satisfaction [10]. In this context, novelty and popularity aspects as well as the listener's music preferences (as part of the user properties – cf. Section 2.1) need to be taken into account when designing a serendipitous system. For instance, a fan of Medieval Folk Metal music will be rather disappointed and likely even bored if the system recommends the band "Saltatio Mortis"[8], disregarding his individual music knowledge and taste. In contrast, for a user occasionally enjoying "Metallica" and "Bob Dylan", the same recommendation may prove serendipitous.

Spatiotemporal Awareness

As already indicated in the description of popularity/familiarity, spatial and temporal considerations are important for intelligent music access systems. However, time and location do not only influence the popularity of a music item, but also the individual user preference for a particular style of music. For instance, a user interested in music, traveling, and learning about new cultures will likely prefer recommendations tailored to his location when going

[7] `http://musicmachinery.com/2009/05/25/artist-similarity-familiarity-and-hotness`
[8] "Saltation Mortis" is one of the major Medieval Folk Metal bands.

abroad. Another person, while preparing to go out on a Saturday evening, will have an entirely different musical entertainment need and hence musical preference than the same person coming home after a long work day.

3 State-of-the-Art and Open Challenges

A literature overview, in particular focusing on the state-of-the-art in topics relevant to this thesis, is given in the following. Furthermore, open challenges are identified, some of which are addressed in the main scientific contributions of this thesis (cf. Chapter 4).

3.1 Multimodal Music Information Retrieval

As defined in Sections 2.1 and 2.2, multimodality in MIR can refer to perceptual features and to access modalities, respectively. As for the former, most MIR research was traditionally carried out in the *music content* domain, i.e., information was derived from the audio signal. Research aimed at describing human music perception via computational models and building applications using these models started in the late 1990s/early 2000s [89, 18, 39, 98]. Although first work on harvesting *music context* sources (web pages, in particular) for MIR tasks emerged in the early 2000s [12] as well, it was not before the mid-2000s when contextual data sources began to be widely used in MIR research. Early work includes [16, 40, 26, 76].

Except for some few works published around the mid-2000s, e.g. [8, 34], which took a more holistic view on MIR, considering multimodality via joining various computational music features is a very recent, but upcoming direction in the field [85]. Several initiatives to foster this kind of multimodality in MIR research emerged during the last couple of years. For instance, the author co-founded the `International Workshop on Advances in Music Information Research` (AdMIRe) series, the first edition of which was held in 2009[9]. AdMIRe is now in its fifth year of existence. It has a focus on hybrid and multimodal approaches to MIR. Also striving to foster user-centric and multimodal MIR, the `International ACM Workshop on Music Information Retrieval with User-Centered and Multimodal Strategies` (MIRUM) started in 2011[10] [38] and continued its success in 2012.

At the same time, some multimodal music datasets and benchmarking initiatives that make use of them emerged. Two of these datasets and the corresponding challenges are presented in the following. The `Million Song Dataset`[11] (MSD) includes editorial metadata and audio features for one million contemporary popular music tracks. Although the MSD has often been criticized for offering rather primitive audio features, its true wealth can be found in the multimodality of data sources. In addition to metadata and music content features, the MSD provides indications of cover songs, bag-of-words representations of song lyrics, collaborative tags, song similarities, and data about individual listening events. The MSD was recently used in a music recommendation benchmarking initiative, the MSD Challenge[12]. Its aim was to predict user ratings of music items in a typical recommendation scenario [43]. Other work that exploits the MSD includes [44], in which McFee and Lanckriet use the dataset to model playlists via random walks, integrating audio features, co-listening data, release era, familiarity, lyrics, and social tags.

[9] `http://www.cp.jku.at/conferences/admire2009`
[10] `http://mirum11.tudelft.nl`
[11] `http://labrosa.ee.columbia.edu/millionsong`
[12] `http://labrosa.ee.columbia.edu/millionsong/challenge`

Another recent initiative is the `MusiClef` series of benchmarking challenges, which the author co-organizes. The task to be solved in its 2011 and 2012 editions, [47] and [46], respectively, was a music auto-tagging task, for which several multimodal data sources were provided, including editorial metadata, different audio descriptors, collaborative tags, multilingual sets of web pages and corresponding term weighting features. As ground truth served professional annotations of the music pieces. The dataset is presented in [78] and is publicly available[13].

Research directed at the latter interpretation of multimodality, i.e. different modalities in music access schemes, is vital to build intelligent music retrieval systems. Such systems strive to offer the user always the suited query formulation facility (be it implicitly[14] or explicitly[15]), adapted to his or her general user properties, current user context, and information or entertainment need.

The state-of-the-art in multimodal music access schemes includes work on *adaptive, personalized, and user-aware MIR* as well as *multimodal user interfaces to music collections*. The former is comprehensively addressed in the PhD thesis of Sebastian Stober [95]; general directions in user-aware MIR are discussed in a publication by the author of this thesis [85]. Also Liem et al. discuss and highlight the importance of user-centered strategies in music retrieval [38]. Furthermore, the research group around Francesco Ricci elaborates personalized and context-aware music recommendation systems. Kaminskas and Ricci approach the problem of suggesting music that fits a place of interest by making use of user-generated tags that describe music pieces and tags that describe places. Investigating different similarity measures defined between the two kinds of tag features, in a user-centric evaluation experiment, the authors found that the Jaccard index performs best to predict music suited for a place of interest [25]. Baltrunas et al. in [5] propose a context-aware music recommendation system for car driving situations. The system takes into account eight different contextual factors (e.g., driving style, mood, road type, weather, traffic conditions). Although results are promising, it has to be critically remarked that their application scenario is quite restricted and their system relies on explicit human feedback, which might not be appropriate in driving situations.

Research on multimodal user interfaces to music collections includes the work by Lübbers and Jarke [41], who present a browsing interface that makes use of a three-dimensional visualization technique to illustrate clusters of music pieces. These clusters are computed based on an ensemble of audio and context features. In collaboration with Peter Knees, Tim Pohle, and Gerhard Widmer, the author of this thesis developed the "nepTune" music browsing and exploration interface [30], an extension of which was selected for the publications constituting this thesis (cf. [70] [G] and Figure 3). The main motivation of the nepTune interface, similar to Lübbers and Jarke's system, is to make music browsing entertaining. To this end, both systems offer the user a game-like navigation interface, and allow him or her to explore a virtual landscape created from the clusters of similar music pieces. More information on how to define this similarity, which is a key task in MIR, is given in Section 3.2. Both systems combine low-level features calculated from different sources; nepTune further integrates high-level information such as descriptive tags and music-related images (e.g., album covers or band photographs).

[13] `http://www.cp.jku.at/musiclef`

[14] An example for an implicit query is the listening history of the user, which can be used in a recommendation system to suggest novel music. Another one is browsing, where the "query" consists of the constant user interaction with the browsing interface.

[15] An example for an explicit query is a traditional metadata-based search query.

When it comes to innovative user interfaces to music collections, the work by Masataka Goto is without doubt one of the most important as he co-developed several such interfaces: "Musicream" [20], "MusicRainbow" [51], and "MusicSun" [52]. While Musicream offers a joyful and easy way to create playlists based on audio similarity, MusicRainbow and MusicSun support multimodal music browsing by providing music content- and music context-centered interaction facilities.

Probably the biggest **open challenges** in the context of multimodal MIR are (i) to understand how the diverse multimodal features that are already available can be integrated to build personalized music retrieval systems, (ii) to research ways to model the user and his or her cultural and environmental context, (iii) to investigate the user's individual information or entertainment need, given his or her user model, and (iv) to automatically determine the user's preferred music access modality and adapt the music retrieval system accordingly, i.e., research how to present the multimodal information to the user in the most beneficial way.

Challenges (i) and (ii) are addressed in publication [83] [H], which is part of this thesis. Here we first investigate different combinations of state-of-the-art music content and music context features and determine an optimal combination of the two for similarity and retrieval tasks. We then look into different ways to incorporate user context features, more precisely, geolocations of users' listening events, to build personalized music recommendation systems.

The latter point (vi) requires novel types of user interfaces yet to develop, as the vast majority of music retrieval systems still rely on metadata-based search. One step towards such intelligent user interfaces is made in publication [70] [G], in which music content and music context features are combined to offer the user a joyful, game-like, multimodal music browser (cf. Figure 3).

The ultimate aim here is to comprehensively combine aspects of both interpretations of multimodality to build holistic systems in which different types of features are used to provide various personalized access schemes to the user. To give an example, by fusing information about audio similarity, regional popularity, and the user's listening history, such a system would be able to recommend music that is acoustically similar to his or her preferred music, but is furthermore adapted to the music trends at his or her current location while traveling.

3.2 Music Similarity Measurement

Similarity estimates between music artists or pieces are a vital building block for many MIR applications as they enable, among others, automatic playlist generation, music recommendation systems, or user interfaces that foster browsing music collections in an intuitive way. A good overview of features and similarity measures can be found in [9] for the music content and in [72] for the music context. As already elaborated on in the previous section, computational music features derived from the audio signal and similarity measures applied to them ignore, however, *individual perception of music similarity*. One remedy to alleviate this issue is incorporating music context and/or user-specific information into the similarity measure. Since it is very hard to gather a decent amount of data on user context and user properties, the use of what is frequently called "community metadata" (i.e., a categorical subset of the music context, introduced in Section 2.1) can be seen as an intermediate step towards personalized similarity measures.

In this context, related work on music context-based similarity measurement can be categorized into (i) *text-based approaches* that employ Text-IR methods and (ii) *co-occurrence-based approaches* that use the co-occurrence of two music items as indicator for their similarity (for instance, computed on playlists or shared folders in peer-to-peer networks). It also seems

reasonable to categorize these approaches according to the data source they exploit: web pages, microblogs, collaborative tags, playlists, peer-to-peer networks, just to mention some. Scientific work that defines the state-of-the-art in each of these categories is briefly presented in the following:

Web Pages:

Web pages are frequently used as source for both kinds of approaches: Text-IR- and co-occurrence-based methods. For the former, a reference work is [80], which is part of this thesis [E]. It is hence presented in Section 4 and included as full text in Part II. In addition, the author also proposed several co-occurrence-based approaches, for instance in [76, 71, 60].

Exploiting web pages as contextual source for MIR has a longer tradition that ranges back to the year 2000, when Cohen and Fan [12] proposed to use term feature vectors from web pages for music artist similarity estimation. They extract lists of artist names from web pages determined by querying web search engines. The resulting pages are then parsed according to their DOM tree, filtered, and sought for occurrences of entity names related to music. Term vectors of *co-occurring artist names* are subsequently used to build a recommendation system. Term vector representations of artists whose term weights are computed as co-occurrence scores is an approach also followed later in [105, 76, 21]. In contrast to Cohen and Fan's approach, Zadel and Fujinaga [105] and Schedl et al. [76] derive the term weights from *search engine's page count estimates* and suggest their method for artist recommendation. A similarity function commonly used in co-occurrence-based approaches (or variants thereof) is given in Equation 1, where $co(A_i, A_j)$ denotes the total number of co-occurrences of artists A_i and A_j in the set of web pages known to mention artist A_i, and $occ(A_i)$ represents the total number of web pages in which the name of artist A_i occurs; analogously for $occ(A_j)$ and $co(A_j, A_i)$.

$$sim(A_i, A_j) = \frac{1}{2} \cdot \left(\frac{co(A_i, A_j)}{occ(A_i)} + \frac{co(A_j, A_i)}{occ(A_j)} \right) \tag{1}$$

Computing term feature vectors from term sets other than artist names, hence following the Text-IR approach, is performed by Whitman and Lawrence [103]. They extract different term sets (unigrams, bigrams, noun phrases, artist names, and adjectives) from up to 50 artist-related web pages obtained via a search engine. After having downloaded the pages, the authors apply parsers and a Part-of-Speech (POS) tagger to assign each word to its suited test set(s). An individual term profile for each artist is then created by employing $tf \cdot idf$ weighting. The overlap between the term profiles of two artists, which is defined by the authors as the sum of weights of all terms that occur in both term profiles, is then used as an estimate of their similarity. Extending the work presented in [103], Baumann and Hummel [6] introduce various filters to prune the set of retrieved web pages (length-based filtering, advertisement filtering, and keyword spotting in the URL, the title, and the first text part of each page). Unlike Whitman and Lawrence, who experiment with different term sets, Knees et al. [26] present a similar approach using only one list of unigrams. For each artist, a weighted term profile is created by applying a $tf \cdot idf$ variant; cosine similarity is used to compute resemblance between these term profiles.

Microblogs:

Mining microblogs to derive music similarity estimates is very sparsely researched so far. To the best of the author's knowledge, most work on corresponding approaches is (co-)authored

by himself. In particular, [64] [F] comprehensively analyzes the use of *Text-IR techniques* to estimate similarities between music artists (and movies) from microposts. Since this work constitutes part of the thesis at hand, its main findings are summarized in Section 4 and the full paper can be found in Part II.

Schedl and Hauger present in [68] a *co-occurrence-based approach* to derive artist similarities from microposts. To this end, they first identify tweets reporting music listening activity by filtering the `Twitter` stream for hashtags such as `#nowplaying`. This yields a set of candidate artists which is subsequently matched against a database of artists and tracks from `MusicBrainz`[16] to filter noise and other events not related to music listening[17]. Schedl and Hauger then analyze which artists or tracks are listened to by the same user and propose several similarity measures based on different combinations of single artist counts and co-occurrence counts.

Another quite similar work is [106], although its focus is on music recommendation, not explicit similarity measurement. Zangerle et al. perform basically the same preprocessing step as Schedl and Hauger to derive `<user,song>`-pairs from microblogs. In contrast to Schedl and Hauger who evaluate different normalization strategies to account for varying popularity of artists that might distort the results of their similarity estimators, Zangerle et al. use the absolute number of co-occurrences to build a music recommender system. Both groups of authors perform evaluation using `Last.fm` similarities as ground truth and compare the overlap between the `Twitter`-based most similar artists and those returned by `Last.fm`, taking as seed each artist in the test collection. The best performing similarity estimator found by Schedl and Hauger is given in Equation 2, where $co(A_i, A_j)$ denotes the total number of co-occurrences of artists A_i and A_j in the tweets of same users, and $occ(A_i)$ represents the count of artist A_i in the entire corpus of tweets.

$$sim(A_i, A_j) = \frac{co(A_i, A_j)}{\sqrt{occ(A_i) \cdot occ(A_j)}} \tag{2}$$

Collaborative Tags:

Collaborative music tags are usually the result of (i) users labeling artists, albums, or songs on platforms such as `Last.fm` or (ii) users playing "games with a purpose" [101] that collect music annotations. Work that mines similarity information from the former source of tags includes [17], in which Geleijnse et al. gather tags from `Last.fm` to construct a "tag ground truth" on the artist-level. The authors first filter redundant and noisy tags using the set of tags associated with tracks by the artist under consideration. Similarity between two artists is then estimated as the number of overlapping tags. Evaluation on a set of 1,995 artists, using `Last.fm`'s similar artists function as ground truth, shows that the number of overlapping tags between similar artists is much larger than the overlap between arbitrary artists (about 10 vs. 4 tags after filtering).

Exploiting collaborative tags to compute music similarity is also the aim of [37], where Levy and Sandler construct a semantic space for music pieces based on tags retrieved from `Last.fm` and from `MusicStrands`[18], a web service (no longer in operation) that allowed users to share playlists. All tags found for a specific music piece are tokenized and a document-term matrix is created based on $tf \cdot idf$ weightings. Each track is hence represented by a term

[16] `http://musicbrainz.org`
[17] For instance, `#nowplaying World of Warcraft` or `i am #nowplaying my favorite tune`.
[18] `http://music.strands.com`

weight vector. Three different approaches are considered to compute the *tf* term: takinng into account the number of users that applied the tag, ignoring the number of users (performing no *tf* weighting at all), and restricting the terms to adjectives by employing a POS tagger. The authors evaluate their approach in a retrieval task and report average precision values. A retrieved term is considered relevant if it is assigned the same genre or artist label as the seed. Levy and Sandler find that using all terms (not only particular linguistic categories such as adjectives) is preferable. So is the incorporation of the number of users that applied the tag into the *tf* score.

The second source of collaborative tags is "games with a purpose" [101]. Such games aim at solving tasks that are hard or infeasible to perform for a computer by means of human power. They obviously have to be entertaining enough to attract and keep many users playing. Games for music tagging include "TagATune" [35], "Listen Game" [97], and a game proposed by Mandel and Ellis in [42]. In particular TagATune is highly related to perceptual similarity measurement, as it not only implements a tagging task, but also "comparison rounds" in which users are presented three songs: one seed track and two alternatives to choose from. Users have to decide which of the alternatives sound more similar to the seed song. From this kind of information, relative similarity judgments and in turn a similarity measure can be derived [36, 95, 104].

Playlists:

Different kinds of playlists are reported in literature to derive music similarity measures: for instance, radio station playlists, compilation CDs, and user-generated playlists. One of the earliest works that exploits playlists is [48], in which Pachet et al. extract co-occurring music items (artists and songs) from a French radio station and from compilation CDs. The authors compute the co-occurrence count of two artists (or pieces of music) A_i and A_j in a playlist. The co-occurrence of an item A_i to itself is defined as the number of A_i's occurrences in the playlist under consideration. These counts are then normalized to account for different frequencies, i.e. popularities, of songs or artists. The similarity measure inferred from the raw counts is given in Equation 1.

Exploiting social media, Baccigalupo et al. propose to derive artist similarity information from playlists shared by members of a web community [3]. To this end, the authors look at more than one million playlists made publicly available by `MusicStrands` (sharing of playlists is no longer operational). The authors extract the 4,000 most popular artists, defining popularity as the number of playlists in which an artist occurs. This definition is equivalent to the document frequency in IR parlance. Taking into account that two artists consecutively occurring in a playlist are presumably more similar than two artists occurring farther away in a playlist, Baccigalupo et al. suggest to use a distance function $d_h(A_i, A_j)$ that counts how often a song by artist A_i co-occurs with a song by A_j at a distance of h, i.e. the number of songs in between occurrences of artists A_i and A_j. To measure the dissimilarity of two artists A_i and A_j, the authors propose Equation 3, in which the playlist counts at distances 0 (two consecutive songs by artists A_i and A_j), 1, and 2 are weighted with factors β_0, β_1, and β_2, respectively. The weights are empirically chosen as $\beta_0 = 1$, $\beta_1 = 0.8$, $\beta_2 = 0.64$. Similar to Pachet et al.'s work [48], Baccigalupo et al. adapt their dissimilarity measure to account for the "popularity bias"[19] according to Equation 4,

[19] The "popularity bias" refers to the fact that very popular music items co-occur with a lot of others, because they are well-known and often listened to by the average music listener. If the similarity measure is not corrected for them, they will likely result in "hubs" [57, 90], a high number of which is

where $\widehat{dist}(A_i)$ denotes the average distance between A_i and all other artists (Equation 5), and X is the set of the $n-1$ artists other than A_i.

$$dist(A_i, A_j) = \sum_{h=0}^{2} \beta_h \cdot [d_h(A_i, A_j) + d_h(A_j, A_i)] \tag{3}$$

$$|dist|\,(A_i, A_j) = \frac{dist(A_i, A_j) - \widehat{dist}(A_i)}{\left| max\left(dist(A_i, A_j) - \widehat{dist}(A_i)\right)\right|} \tag{4}$$

$$\widehat{dist}(A_i) = \frac{1}{n-1} \cdot \sum_{j \in X} dist(A_i, A_j) \tag{5}$$

Peer-to-peer Networks:

Peer-to-peer (P2P) networks in which users share different kinds of data are another source to harvest music-related information. Users commonly provide metadata about their shared content, in the case of music, typically file names and ID3 tags. Based on artist or track co-occurrences in a user's shared folder, it is possible to create music similarity measures.

Early work that makes use of data extracted from P2P networks includes [16, 103, 40, 7]. All of these authors extract data from the P2P network `OpenNap` and subsequently derive music similarity information. Logan et al. [40] and Berenzweig et al. [7] retrieved the 400 most popular artists in `OpenNap` as of mid 2002. Gathering metadata on shared content yielded about 175,000 user-to-artist relations from about 3,200 music collections. Logan et al. compare similarities defined by artist co-occurrences in `OpenNap`, by expert opinions from `Allmusic`[20], by playlist co-occurrences from `Art of the Mix`[21], by data gathered from a web survey, and by MFCC-based audio features [2]. Using an overlap score among the most similar artists given by each similarity measure under consideration, the authors find (i) that co-occurrence data from `OpenNap` and from `Art of the Mix` shows a high degree of overlap, (ii) that the experts from `Allmusic` and the participants of the web survey agree moderately, and (iii) that the MFCC-based similarity measure exhibits low agreement with all other sources.

In [103] Whitman and Lawrence retrieve from `OpenNap` a total of 1.6 million user-song relations over a period of three weeks in August 2001. The authors address the popularity bias by using the similarity measure shown in Equation 6, where $uc(A_i)$ denotes the number of users that share songs by artist A_i, $uc(A_i, A_j)$ is the number of users that have both artists A_i and A_j in their shared collection, and A_k is the most popular artist of the whole dataset. The second factor (in the right hand part of the equation) downweights the similarity between two artists if one of them is very popular and the other is not.

$$sim(A_i, A_j) = \frac{uc(A_i, A_j)}{uc(A_j)} \cdot \left(1 - \frac{|uc(A_i) - uc(A_j)|}{uc(A_k)}\right) \tag{6}$$

More recently, Shavitt and Weinsberg derived similarity information on the artist- and song-level from the `Gnutella` file sharing network [93]. The authors collected metadata of

usually undesirable for retrieval and recommendation tasks.

[20] `http://www.allmusic.com`

[21] `http://www.artofthemix.org`

shared files from more than 1.2 million `Gnutella` users in November 2007. They restricted their search to mp3 and wav files. The crawl yielded metadata for about 530,000 songs. Information on both users and songs are then represented via a 2-mode graph. A link between a song and a user is created if the user shares the song. It turns out that most users of the P2P network share similar files. In addition, Shavitt and Weinsberg address the problem of song clustering. Accounting for the popularity bias, the authors define a distance function that is normalized according to song popularity, as shown in Equation 7, where $uc(S_i, S_j)$ denotes the total number of users that share songs S_i and S_j. c_i and c_j denote, respectively, the popularity of songs S_i and S_j, measured as their total occurrence in the dataset.

$$dist(S_i, S_j) = -\log_2 \left(\frac{uc(S_i, S_j)}{\sqrt{c_i \cdot c_j}} \right) \tag{7}$$

Some of the biggest **open challenges** in the context of music similarity measurement, according to the author, are to understand (i) how the low-level and mid-level music content and music context features relate to human music perception[22] and (ii) how to use this knowledge to construct multifaceted similarity measures that reflect human perception of similarity.

Although these challenges are not comprehensively addressed by the author's publications included in this thesis, (i) is targeted in [92], where we propose an algorithm to infer semantic tags from audio features. In [91] an approach that uses tags and audio features to improve music similarity measures is presented. In [69] the author proposes a multifaceted music similarity measure that takes user properties and user context factors into account. Publication [83] [H] of this thesis investigates combinations of music content, music context, and user context features and similarity information, aiming at building a personalized and context-aware music recommendation system. This paper hence represents one step into solving challenge (ii).

3.3 Music Information Extraction

Approaches to automatically extract or infer different pieces of music-related information are rather sparsely proposed in literature, except for the task of *auto-tagging*, which has recently become quite popular. It is the process of automatically labeling music pieces with semantic tags. An overview of the state-of-the-art in music auto-tagging can be found in [65].

Given the scope of this thesis, i.e. MIR and Music Information Extraction (MIE) from web and social media sources, a comprehensive elaboration on web-based MIE approaches can be found in the PhD thesis of the author [60]. Approaches to extract or infer different pieces of music-related information are presented. This information is subsequently used to build a music information system. Since a shortened version of [60] forms part of this thesis, publication [87] [B], we will elaborate on the respective approaches in Section 4.

Among the few scientific works on web-based MIE is [28], in which Knees and Schedl investigate two approaches, supervised learning and rule-based pattern extraction, to derive two different kinds of relations from music-related web pages. These relations are *members of a band* and *artist discographies*. The rule-based approach resembles the one presented in [86], in which patterns such as "M plays the I" or "M is the R of B" are used to identify

[22] To answer this question, we first need to investigate whether there are relations that are generally valid, independent of individual and culture, or if perception of music is too individual to derive such patterns.

members (M), instruments (I), and roles (R) in bands (B). The supervised learning approach uses Support Vector Machines (SVM) [100] on features representing POS tagging results, gazetteer-based entity information, identified person entities, and the textual surrounding of each token. Experimental results revealed a superiority of manually crafted rules over automatic approaches.

Web-based approaches to predict the *country of birth (for artists)* or *country of origin (for bands)* are investigated in [84, 82] by the author of this thesis and in [22] by Govaerts and Duval. In our work we propose three heuristics: (i) page count estimates returned by the `Google`[23] search engine for queries of the form `"artist/band" "country"`, (ii) $tf \cdot idf$ weighting of country names in web pages mentioning the artist/band under consideration, and (iii) text distance between country names and key terms such as "born" or "founded" in the same set of web pages as used in approach (ii). We find that simple tf outperforms heuristics (i) and (iii), but also the $tf \cdot idf$ weighting. We presume that the underperformance of $tf \cdot idf$ is due to the idf term that downweights too heavily common country names and country abbreviations[24] such as "US".
Govaerts and Duval's work [22] differs from ours in several regards. They use a fixed set of specific web pages such as `Last.fm`, `Wikipedia`[25], and `Freebase`[26] to gather music-related pages. The authors extract artist biographies from these pages and propose simple heuristics to determine the artist's country of origin. One of the used heuristics is predicting the country that is most frequently mentioned in the biographies of the artist under consideration. Another one is favoring country names that occur early in the biographies. Some `Wikipedia` and `Freebase` pages explicitly mention country names in an "origin" attribute. This information is also used by Govaerts and Duval for country prediction. Not too surprisingly, such explicit mentions of country names performed best in terms of precision, although at low recall values. To increase coverage, the authors also propose a fusing strategy of the classification heuristics.

The emergence of social media also provided an unprecedented source for methods that estimate *popularity* of all kinds of objects and subjects, ranging from politicians [13] to user-generated videos [96]. Since music is comparable to movies in terms of business value, many parties show interest in good estimates for the popularity of music items (in particular, song/album releases) and artists (for instance, songwriters or performers). It hence does not come as a surprise that there are quite a few companies that established their business on this very task, `Musicmetric`[27] and `Media Measurement`[28], just to mention two.
On the scientific side, [81] [C] presents a study on approximating the popularity of music artists using different data sources: (i) search engine page counts, (ii) occurrences in microblogs, (iii) occurrences in shared folders of P2P network users, and (iv) play counts of `Last.fm` users. We analyze whether the popularity rankings provided by the different data sources correlate. Most pairs of approaches show only weak correlation. We attribute this finding to the fact that music popularity is multifaceted and that different data sources reflect different aspects of popularity. On the other hand, P2P rankings and page count rankings reveal some correlation. This can be explained by the accumulating nature of both data sources, i.e., unlike dynamic sources such as `Twitter` streams and traditional music charts,

[23] `http://www.google.com`
[24] We use lists of country synonyms and abbreviations in the indexing process.
[25] `http://www.wikipedia.org`
[26] `http://www.freebase.com`
[27] `http://www.musicmetric.com`
[28] `http://www.mediameasurement.com`

shared folders and page counts are data sources that typically change much less over time.

As Music Information Extraction is still in its infancy, the most important **open challenge** in this context is probably to increase the correctness of the extracted pieces of information. This can be achieved in several ways, for instance, by (i) improving methods to resolve ambiguities of music-related items (e.g., band names that equal common speech terms, such as "Kiss", "Bush", or "Porn"), (ii) computing some measure of confidence in or trustworthiness of different data sources (e.g., the web page of an expert-based music information system is likely to offer more accurate information than the `Twitter` post of a single user), or (iii) employing novel techniques to gather different music-related pieces of information (e.g., via crowdsourcing or "games with a purpose" [36]).

As part of this thesis, publication [87] [B] addresses in detail the task of MIE from web sources and proposes different approaches to infer artist similarity, artist prototypicality for a genre, descriptive labels, band members and instrumentation, and images of album cover artwork. Alleviating the issue of ambiguities of artist names, [77] proposes the use of a penalty term in the context of artist prototypicality estimation. However, approaches that generalize to a wider range of MIR and MIE tasks still need to be researched.

4 Scientific Contributions

The theme of this postdoctoral thesis is the exploitation of **multiple data sources for music retrieval tasks**, in particular focusing on web and social media sources. Furthermore, the **combination of music content, music context, and user context** features to build **intelligent multimodal music access systems** is addressed too.

Table 1 gives an overview of some of the author's publications related to the areas of multimodal music access, social media mining, similarity measurement, information extraction, popularity estimation, auto-tagging, and evaluation. Some survey articles are included as well. As space limitations render impossible to include in this thesis all publications by the author, the ones shown in Table 2 were selected to constitute the core part of this thesis. The full publications can be found in Part II and are summarized in Section 4.1. The corresponding main scientific contributions of the thesis are highlighted in Section 4.2.

4.1 Summary of Core Publications

Publication [A] highlights the importance of *user-centric evaluation* in MIR, which is not adequately addressed in most recent publications of the field. The article presents a discussion of MIR literature that includes only systems or virtual users in the evaluation process. On the other hand, also the few examples of papers that provide experimentation involving real users are critically investigated. We strongly advocate to conduct truly user-centric experiments in MIR, as already common in other fields such as recommendation systems. Publication [A] also provides suggestions on how to perform user-centric evaluation experiments. We eventually show that human effort can be considerably reduced when making use of novel evaluation strategies [99].

Publication [B] presents a set of Web Mining and Information Extraction techniques adapted to the music domain. A music information system whose database is automatically populated by applying these techniques is proposed and evaluated on a real-world collection of over half a million of artists. The pieces of musical information extracted from the web include *similarity between artists*, *band members and instrumentation*, *album cover artwork*, *descriptive labels*, and *"prototypicality"* of an artist or band for a genre.

Table 1 Selected publications by the author, related to the scope of this thesis.

Subject Matter	**Publications**
General Considerations on Multimodal and Personalized Music Retrieval	[85, 69, 73, 72]
Multimodal Retrieval and Browsing	[83, 78, 24, 70, 27]
Social Media Mining	[64, 23, 68, 66]
Similarity Measurement	[64, 68, 74, 80, 91]
Information Extraction	[87, 84, 82, 88]
Artist Popularity Estimation	[62, 81, 68, 66]
Auto-tagging	[78, 91, 61, 92]
Evaluation Studies	[46, 64, 80]

Table 2 Publications constituting the core part of this thesis.

[A] Markus Schedl, Arthur Flexer, Julián Urbano. **The Neglected User in Music Information Retrieval Research**. *Journal of Intelligent Information Systems*, 2013.
[B] Markus Schedl, Gerhard Widmer, Peter Knees, Tim Pohle. **A Music Information System Automatically Generated via Web Content Mining Techniques**. *Information Processing & Management*, 47, 2011.
[C] Markus Schedl, Tim Pohle, Noam Koenigstein, Peter Knees. **What's Hot? Estimating Country-Specific Artist Popularity**. In *Proceedings of the 11th International Society for Music Information Retrieval Conference* (ISMIR), Utrecht, the Netherlands, August 2010.
[D] Markus Schedl. **Leveraging Microblogs for Spatiotemporal Music Information Retrieval**. In *Proceedings of the 35th European Conference on Information Retrieval* (ECIR), Moscow, Russia, March 2013.
[E] Markus Schedl, Tim Pohle, Peter Knees, Gerhard Widmer. **Exploring the Music Similarity Space on the Web**. *ACM Transactions on Information Systems*, 29(3), July 2011.
[F] Markus Schedl. **#nowplaying Madonna: A Large-Scale Evaluation on Estimating Similarities Between Music Artists and Between Movies from Microblogs**. *Information Retrieval*, 15:183–217, June 2012.
[G] Markus Schedl, Christian Höglinger, Peter Knees. **Large-Scale Music Exploration in Hierarchically Organized Landscapes Using Prototypicality Information**. In *Proceedings of the ACM International Conference on Multimedia Retrieval (ICMR)*, Trento, Italy, April 2011.
[H] Markus Schedl, Dominik Schnitzer. **Hybrid Retrieval Approaches to Geospatial Music Recommendation**. In *Proceedings of the 35th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, Dublin, Ireland, July–August 2013.

Publication [C] proposes and evaluates various approaches to estimate the *popularity* of music artists or bands. These approaches rely on data gathered from social media or related sources (microblogs, shared listening histories, and peer-to-peer networks, for instance). Using spatial information where possible allows to create "Social Media Charts" for each country of the world, provided enough data is available. The publication further discusses different types of popularity; for instance, "hotness" that relates to recent buzz of an artist, or "familiarity" that measures the overall number of people who know the artist – cf. Section 2.2. Popularity information nicely complement the other types of information mined from the web in publication [B]. Whereas [B] focuses on static web pages, [C] foremost exploits social media.

In publication [D] the author extracts *spatial and temporal information about music listening events* from microblogs. These are subsequently analyzed with respect to their temporal stability and spatial distribution of music preference. It is found that global music taste does not significantly differ between months of the year (keeping location fixed), but it does between workdays and weekends. The dataset of annotated listening events is made available to the public and constitutes the first freely available corpus of music listening information inferred from microblogs. Due to the availability of user identifiers, temporal, spatial, and musically descriptive labels in the dataset, the author expects the set to be used for various user context-aware music retrieval tasks.

Publications [E] and [F] profoundly address *evaluation of similarity measures* derived from music context data sources. More precisely, in [E] a large-scale investigation of various aspects in modeling artist-related documents from web pages, term weighting features, and similarities computed between these documents/features is conducted. Publication [F] reports on a similar study carried out on microblog data, this time not restricted to the music domain, but further investigating the use of social media data to predict the genre of movies. Summarizing the main findings of the two sets of experiments, it was shown that using a domain-specific dictionary to index the documents (web pages or microblogs) yields more stable and overall better results than computing term weight vectors from the entire set of terms appearing in the corpus. Dictionaries of music-related terms also proved to outperform general English dictionaries typically used in spell checking applications. Another finding is that restricting the query used for document selection by additional keywords (typically "music") is preferable for web pages, but deteriorates performance of microblog-based music similarity algorithms. This is due to the fact that microposts are restricted to 140 characters and are hence usually written in a concise and non-redundant way. In terms of term frequency (*tf*) and inverse document frequency (*idf*) formulations, results are inconclusive regarding the choice of a particular term weighting function. Logarithmic formulations for both the *tf* and the *idf* tend to outperform other variants, though, regardless of the data source. As for the actual similarity computation between documents, microblogs seem to benefit from a simple inner product measure without any kind of normalization, while web pages require the use of standard cosine similarity or Jaccard coefficient.

Publication [G] proposes the multimodal music browsing interface "deepTune", which extends the "nepTune" interface presented in [29]. Taking as input an arbitrary digital music collection, deepTune extracts and combines music content features inferred from the audio signal and music context data mined from the web. Constructing a similarity measure and applying clustering techniques, the music collection is then visualized as a virtual three-dimensional music landscape with oceans, beaches, grasslands, mountains, and valleys serving as metaphor for the density of music items in different regions of the map. An example is shown in Figure 3. The generated virtual landscape, which is unique for each

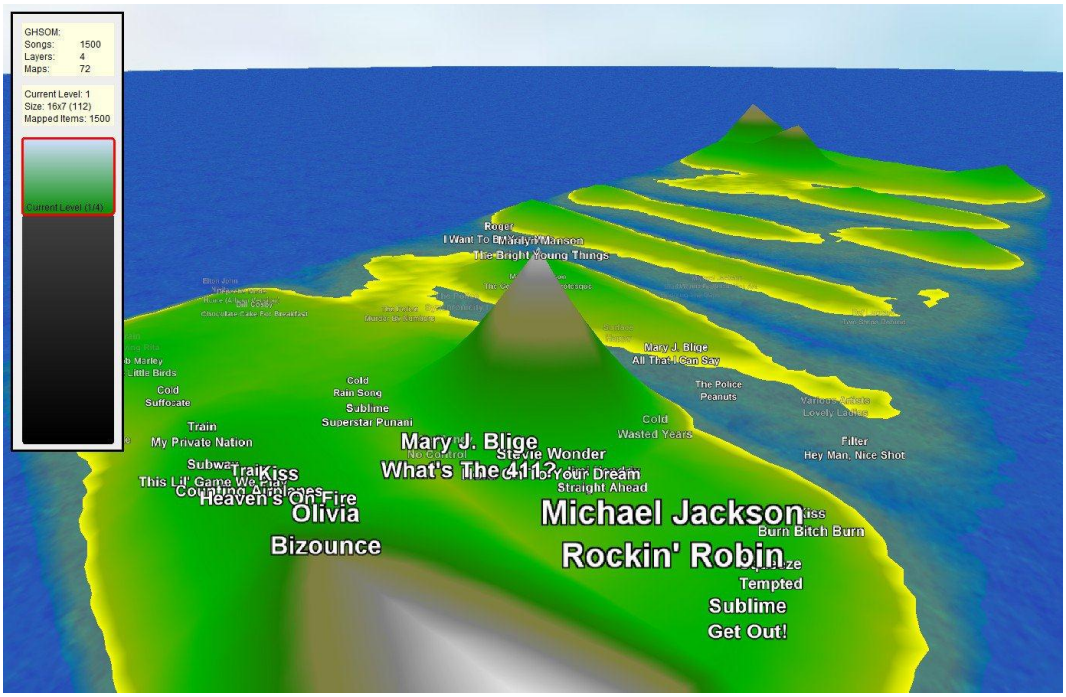


Figure 3 The "deepTune" user interface to explore music collections.

music collection, can be explored in a manner similar to common flight simulator games. Navigating through the music landscape, the user hears the music pieces closest to his current position in the virtual landscape. Employing a hierarchical clustering and visualization model, deepTune (unlike nepTune) is capable of dealing with huge music collections.

Publication [H] investigates different strategies to integrate music content, music context, and user context into a hybrid music recommendation system. Building upon state-of-the-art feature extractors to determine music similarity based on audio content and on web information, we propose novel geospatial music recommendation approaches using location information of microblog users. In a quantitative evaluation experiment, we find that collaborative filtering outperforms music content-based approaches when the dataset contains a high number of users. The opposite is true for datasets with a small number of users. For very active users, including geospatial information in the recommendation algorithm is capable of heaving performance above the levels reached by pure collaborative filtering.

4.2 Main Scientific Contributions

The most important contributions and findings of this thesis, as elaborated on in the selected core publications, are the following:

(I) a critical investigation of the largely neglected role of the user in current research on Music Information Retrieval [A],

(II) novel techniques to derive semantic information from the web and social media, related to music items [B,C], music listening activity of users [D,H], and music preferences of entire populations [C,D],

(III) comprehensive investigations of models to infer music similarity from the web and from social media [E,F] and to combine these context-based similarities with audio-based ones and with user context aspects, and
(IV) a system to explore music collections, which combines music content and contextual data [G].

(I) A critical investigation of the largely neglected role of the user in current research on Music Information Retrieval:

Considering the end user in designing and evaluating music retrieval algorithms and systems should be key, but unfortunately is not in almost all works on MIR. In practice, MIR research usually shows a systems-based evaluation design, i.e., laboratory experiments exist solely in a computer; for instance, evaluation of algorithms is conducted on digital databases. Only very rarely the user is taken into account in a comprehensive manner.

In publication [67] [A], we discuss this systems-based approach, show its shortcomings, and analyze how other communities address the user in design and evaluation of recommendation and retrieval algorithms. The major criticisms concerning the neglected role of the user in MIR research and possible ways to overcome these limitations are the following:

- Frequently, an objective "ground truth" is assumed, against which music retrieval algorithms are evaluated. For instance, genre labels or similarity information (based either on annotations by music experts or the "power of the crowds") are considered a golden standard. Even though these information may have been generated by end users, experimental settings typically ignore the individual and subjective perception of music. Examples are given in [67] [A].
- Existing systems and algorithms very rarely take comprehensive, multimodal, personalized, and context-ware points of views when it comes to model human music perception. We hence propose a novel model that represents music items by their "music content" and "music context", as well as listeners by their static "user properties" and dynamic, context-aware "user context". Examples can again be found in the article in Part II.
- The MIR community should investigate what it can learn and borrow from other communities. In particular, the Recommendation Systems community is quite active in user-centric design and evaluation. There exist quite a few works that propose comprehensive user-centric evaluation strategies for recommendation systems which could easily be adopted in MIR, e.g. [56, 14].
- Analyzing the results of the MIREX[29] "Audio Music Similarity and Retrieval" task, we show that the differences in human judgments of music similarity are larger than the performance gap between the best and the worst algorithms (cf. Figure 2 in [67] [A]). This remarkably demonstrates the demand for user-centric algorithms when it comes to modeling music similarity.
- When it comes to evaluating MIR algorithms in a user-centric manner, we propose to take into account any factor that is able to influence the dependent variable to be assessed in an evaluation experiment (for instance, accuracy or precision). In systems-based evaluation, it is relatively easy to control all important factors, because the experiments are conducted solely in a computer, not in the real world. In a user-centric evaluation,

[29] http://www.music-ir.org/mirex

in contrast, it becomes extremely difficult, if not infeasible, to model all independent variables. Nevertheless, we should consider as many external factors as possible, even at higher costs. Only this way can we increase user satisfaction, not only systemic performance measures.

(II) Novel techniques to derive semantic information from the web and social media, related to music items:

Semantic musical information that helps people find music or explore music collections constitutes an additional asset over purely content-based data. The automated extraction of respective pieces of information is hence an important task in MIR research. These semantic information typically fall into the categories of music context or user context (cf. Figure 1). Corresponding Music Information Extraction (MIE) techniques are presented in several publications included as part of this thesis. The main achievements, organized by type of information, are the following:

Descriptive Labels: Automatically determining semantic labels that describe music items or artists is an important task, in particular to improve semantic search [27] and to enrich music browsing interfaces such as the one proposed in publication [70] [G]. Based on our earlier work [79], publication [87] [B] proposes to use a dictionary of musical terms [50] to mine music-related web pages for descriptive labels. Different term weighting functions are evaluated for this task: term frequency (tf), document frequency (df), and $tf \cdot idf$. We show in a user study that simple df weighting outperforms standard $tf \cdot idf$. Also tf outperforms $tf \cdot idf$. Even though $tf \cdot idf$ weighting performs well in many retrieval tasks [4], it is seemingly not well-suited to determine the most descriptive terms for a given music artist. An explanation for this is the fact that $tf \cdot idf$ is designed to highly value terms that are specific to a certain document (or artist in our case), thus showing high tf but low df scores. This is well desired when it comes to distinguishing one artist from another, but not when it comes to finding descriptive terms.

Prototypicality: In [87] [B] we propose a novel technique to estimate the "prototypicality" of an artist for a genre. To this end, we make use of co-occurrence information extracted from artist-related web pages. Given two artists and one genre, our approach is based on the assumption that the artist less prototypical for the genre occurs more frequently in web pages of the artist who is more prototypical. Take for instance the genre Metal. It is reasonable that a "long-tail" artist such as "Ensiferum" occurs seldom on web pages of well-known artists such as "Metallica". On the other hand, it is likely that "Metallica", a prototypical band for the genre, is mentioned on "Ensiferum"'s pages. We formalize this assumption using the concepts of *forward links* and *backlinks*, similar to `Google`'s Page Rank algorithm [49]. We further introduce a correction factor that penalizes artists whose prototypicality is exorbitant for all genres, to alleviate the problem of artist names being highly ranked due to their resemblance to common speech words, for instance "Kiss" or "Bush". We already showed in [77] that this penalization term considerably improves the prototypicality ranking. The formula used to rank an artist i with respect to a genre g is given in Equation 8, the penalization term in Equation 9, and the backlink/forward link definitions in Equations 10 and 11, respectively. In these formulas, A_g is the set of artists in genre g, A is the set of all artists, I (J) is the set of web pages gathered for artist i (j), and $df_{j,I}$ ($df_{i,J}$) is the document

frequency of artist j (i) in the set I (J).

$$r(i,g) = \frac{\sum_{j \in A_g}^{j \neq i} bl(i,j)}{\sum_{j \in A_g}^{j \neq i} fl(i,j) + 1} \cdot penalty(i) \tag{8}$$

$$penalty(i) = \left\| \log \left(\frac{\sum_{j \in A}^{j \neq i} fl(i,j) + 1}{\sum_{j \in A}^{j \neq i} bl(i,j) + 1} \right) \right\|^2 \tag{9}$$

$$bl(i,j) = \begin{cases} 1 & \text{if } \frac{df_{j,I}}{|I|} < \frac{df_{i,J}}{|J|} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

$$fl(i,j) = \begin{cases} 1 & \text{if } \frac{df_{j,I}}{|I|} \geq \frac{df_{i,J}}{|J|} \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

Album Cover Artwork: Building upon our previous work [75], in publication [87] [B] we propose an approach to automatically detect images of album covers on web pages. To create a corpus of relevant web pages, the suggested hybrid approach employs the query scheme `"artist name" "album title" cover` to query a search engine. Subsequently, a word-level index [107] including HTML tags is constructed. From this index, we extract all `<img>` tags and fetch the corresponding image files. Since preliminary experiments showed that band photographs and images of scanned compact discs are frequently mistaken for album cover images [75], we apply two content-based filters. To address the former category of errors, we filter all images that are not quadratic (within a tolerance range of 15%). We then use a circle detection technique to discard scanned disc images. To this end, we examine small rectangular regions along a circular path that is touched by the image borders tangentially. Analyzing the contrast between subareas of these regions via RGB histograms reveals with high confidence whether the image at hand is in fact a scan of a compact disc, hence discarded. Applying these two filtering strategies yields a set of candidate images. More details on the implementation of these content-based filters can be found in [87] [B].
To obtain a final set of images which our algorithm predicts as album cover images, we use a text-based distance function on the web pages that contain the candidate images for a given artist. More precisely, we rank each potential image according to the distance between its `<img>` tag and the artist name and between its `<img>` tag and the album name. We then select the image with minimal distance to artist and album name, as shown in Equation 12, where $pos_i(t)$ denotes the position i of term/tag t in a given web page of the artist under consideration.

$$min_{i,j,k} \quad |pos_i(< \texttt{img} >) - pos_j(artist)| + |pos_i(< \texttt{img} >) - pos_k(album)| \tag{12}$$

On a test collection of 255 albums by 118 American and European artists, our approach achieves a precision of up to 89% at a recall level of 93%. On a more challenging collection of 3,311 albums by 1,593 artists from all over the world, the approach yields precision values of up to 73% at 80% recall.

Members and Instrumentation of Bands: Analyzing again web pages found by querying a search engine, we aim at extracting the members of a given band as well as their roles, i.e., the instruments they play. In publication [87] [B], and further elaborated in [86] and [88], we investigate different query schemes and find `"band" music members` to perform best. Using this scheme to obtain a set of web pages is hence the first step in the proposed approach. We then extract all 2-, 3-, and 4-grams whose components consist of more than a single character and whose first letter is written in upper case. Using an English dictionary,

we filter n-grams that equal common speech terms. This cascade of filtering steps, which can be regarded as a Named Entity Detection (NED) approach, is essential to suppress irrelevant n-grams and yields a set of potential band members. Subsequently, we use a pattern extraction technique to obtain the instruments played by the potential band members. Seven patterns such as `"M, the R"` or `"M plays the I"`, where `M` is the potential band member, `I` is the instrument, and `R` is the member's role in the band are considered. For `I` and `R`, we use lists of synonyms to cope with the use of different terms for the same concept (e.g., "drummer" and "percussionist"). We then compute the number of occurrences, i.e. the document frequency, of each pattern and accumulate them over all seven patterns for each `<M,I>`-tuple. To suppress uncertain tuples, we filter out those `<M,I>`-pairs whose document frequency is below a dynamic threshold t_f, which is parametrized by a constant f. t_f is expressed as a fraction f of the highest document frequency of any `<M,I>`-pair for the band under consideration. The `<M,I>`-pairs remaining after this final filtering step are predicted. Evaluating our approach on a set of 51 bands with 499 current and former members, we find that (i) query schemes `"band" music` and `"band" music members` outperform other schemes, (ii) the upper limit for the achievable recall is around 50% (because not all band members given in the ground truth actually occur in the set of web pages), (iii) f values in the range $[0.2, 0.25]$ perform best, and (iv) precision values of 43% at 36% recall for current band members and of 61% at 26% for current and former members can be reached[30].

Country of Origin of Artists or Bands: In publications [84] and, in greater detail, in [82] we approach the problem of determining the country of origin of artists (where they were born) and bands (where they were founded). We investigate three text-based heuristics on web pages related to artists or bands: (i) search engine's page counts for queries like `"artist/band" "country"`, (ii) term weighting scores when querying the set of web pages for the artist/band of interest, using each country name (and its synonyms) as query, and (iii) text distance between country names and keywords indicating origin, for instance, "born", "founded", "origin". Given an artist or a band a, the first heuristic predicts the country with highest page count estimate, the second one the country with highest term score (tf, df, or $tf \cdot idf$) when querying the set of a's web pages, and the last heuristic predicts the country with overall minimal distance at character level between any occurrence of the country name and the artist/band name on a's web pages[31].
On a dataset of 578 artists and bands originating from 69 different countries, we find that simple tf weighting yields the best results (F_1 score of 83%). Using page count or text distance as heuristic, in contrast, performs remarkably inferior (maximum F_1 score of 38% and 54%, respectively). Synonyms for country names improve results statistically significantly when including them in the term weighting approaches.

Country-specific Popularity: Publication [81] [C] looks into popularity estimation of music artists in each country of the world. To this end, we make use of various web and social media sources: (i) page count estimates of search engines, (ii) microblogs, (iii) P2P networks, and (iv) `Last.fm`. In order to highlight artists who are particularly popular in the country of interest c and suppress artists who are popular everywhere, we propose a $tf \cdot idf$-like artist weighting scheme. More precisely, given a set of artists A and countries C, the popularity score of an artist a for a country c is computed as shown in Equation 13, where $occ(a, c)$ is

[30] Given that a predicted `<M,I>`-pair is only considered correct if both the band member and her role are correct, these results are quite promising.
[31] We investigated aggregation functions other than the minimum, but using the minimum for both distances (within one of a's pages and over all of a's pages), turned out to perform best in our task.

the occurrence count of a for c as given by the respective data source and df_a is the number of countries in which artist a is known according to the data source, i.e., the number of countries with $occ(a, c) > 0$.

$$p(a, c) = occ(a, c) \cdot \log_2 \left(1 + \frac{|C|}{df_a}\right) \tag{13}$$

When exploiting page count estimates, $occ(a, c)$ is defined as the estimated number of web pages containing both query terms, the artist and the country name, i.e., using as query `"artist" "country"`. For geolocalized microblogs including the hashtag `#nowplaying`, $occ(a, c)$ is given by the occurrence count of a in tweets localized in c. Extracting information on shared folders in P2P networks [31], $occ(a, c)$ is computed as the total number of shared folders whose sharers are located in c according to their IP address. For the ultimate data source, `Last.fm` scrobbles, we simply use the aggregate playcount for each artist a of country c's most active `Last.fm` users.

Since there is no ground truth of country-specific artist popularity, we compare the rankings yielded by the four data sources, on a collection of more than 200,000 artists extracted from `Last.fm`. To this end, we compute the top-n artist rank overlap over all countries, where the rankings are given by the two data sources to compare. Details of the evaluation procedure can be found in publication [81] [C]. We find that on average the rank overlap between approaches is rather low. The highest overlap of 0.67 is achieved between P2P network data and page counts estimates. The small overlap can be explained by the very different nature of the data sources investigated: each is prone to different biases[32], has a different time dependency[33] and different coverage[34]. This finding strongly supports a multimodal point of view on the problem of artist popularity estimation.

Spatiotemporal Music Listening Activities: In publication [66] [D] we propose a method to extract listening events from microblogs, using related hashtags such as `#nowplaying` or `#itunes`. This work resulted in `MusicMicro`[35], the first publicly available dataset of music listening behavior mined from microblogs. A detailed analysis of `MusicMicro`, which covers in its current version listening activity on the Microblogosphere from November 2011 to September 2012, reveals that music taste (assessed via mood tags mined from `Last.fm`) strongly varies between countries. Moreover, investigating the temporal stability shows a significant difference in listening behavior between workdays and weekends, irrespective of the country. In contrast, no significant difference can be made out between months of the year.

In [23] we use an extended version of the `MusicMicro` dataset presented in [66] [D] to investigate music popularity on the level of songs or pieces. Figure 4 reveals the popularity of songs by "Madonna" on the Microblogosphere, over a time period of nine months. It depicts the listening events for each song, aggregated in bins of one week. It can be seen that songs like "Like a Prayer" and "La isla bonita" are hits, which does not come as a surprise. But we can also make a more interesting observation. Starting in the third week of February 2012, the song "Girl Gone Wild" is becoming quite popular, skyrocketing in

[32] For instance, users of `Last.fm` are known to have a music taste different from the overall popularity [33].

[33] While P2P network data and web page indexes are accumulating data sources, thus reflect the *familiarity* of an artist, approaches based on microblogs as well as traditional music charts measure artist *hotness* for a certain period of time (cf. Section 2.2).

[34] While page count estimates and `Last.fm` data is available for almost all 240 countries in the world, spatial coverage of P2P network data is rather low (only 86 countries). However, traditional music charts are available in much fewer countries still.

[35] `http://www.cp.jku.at/musicmicro`

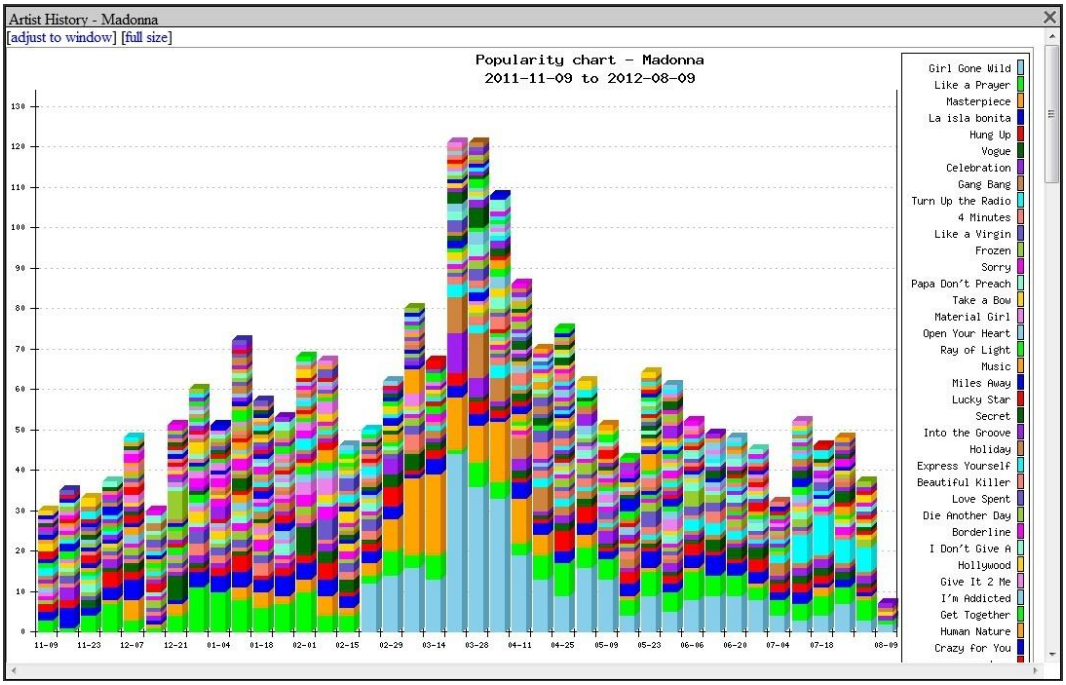


Figure 4 Popularity of songs by "Madonna" over time.

the end of March. The reason for this is seemingly the album release of "MDMA". What is particularly noteworthy, however, is that the album release took place on March 26, i.e., four weeks after the initial appearance of the song "Girl Gone Wild" in the microblog data. Such indications of microbloggers can be explained by official pre-releases of songs or by leakage of songs through channels other than the official album release pipeline. This analysis allows to create "Social Media Charts", similar to the ones investigated in the previous paragraph on country-specific popularity estimation. These are not only helpful for the music industry, but also to build personalized music retrieval and recommendation systems [83].

Mainstreaminess of a Population's Music Taste: Based on such music listening indications shared on the Microblogosphere, we can further assess the "mainstreaminess" of a country's population, provided location information is available for the microblogs under consideration. To this end, in [68] we first aggregate the music artists found in a corpus of microblogs on the level of genres. The assignment between artists and genre is made based on `Allmusic`'s 18 major genres. We then define the *listening pattern* for a city or country c as the relative frequency music of each genre is listened to by users located in c. The elements of the 18-dimensional genre distribution vector $\boldsymbol{g}^c$ for a city or country c are hence computed as shown in Equation 14, where G_i denotes the set of artists assigned to genre i, $occ(a, c)$ is the number of microblogs indicating listening behavior of artist a in city or country c, and A is the set of all artists in the corpus. We use the relative frequency to account for different intensities of microblogging activity in different cities or countries. Nevertheless, we ignore cities or countries with too little coverage to derive reliable listening patterns, i.e., we require at least 100 (artist,user)-pairs in the dataset to make a prediction.

$$g_i^c = \frac{\sum_{a \in G_i} occ(a, c)}{\sum_{a \in A} occ(a, c)} \qquad i = 1 \ldots 18 \tag{14}$$

In order to investigate to which extent the listening patterns differ among different cities or countries c, we calculate the standard deviation of their genre distribution vectors σ^c over all genres (in relation to the global genre distribution, taking the arithmetic mean over the 18 dimensions to obtain a single value). This enables to determine the most and the least representative — or mainstreamy — twittering populations with respect to the average global music listener. The countries whose populations are found to be most and least "mainstreamy" are illustrated in Figures 5 and 6, respectively. The y-axes indicate the relative difference in listening to each genre (from the global average). For instance, Easy Listening seems to be quite popular in Greece, where the amount of music in this genre exceeds the global consumption by a factor of more than three (cf. Figure 5). Confirming a stereotype, Reggae music seems to be extraordinarily popular in Jamaica, being listened to about 1,600% more frequently than on a worldwide scale.

(III) Comprehensive investigations of various models to infer music similarity information from the web and social media:

Publications [80] [E] and [64] [F] harvest web pages and microblogs, respectively, to thoroughly investigate text-based approaches for music artist similarity estimation, a vital ingredient to music retrieval and recommendation systems. Both publications use a similar evaluation framework, in which several thousand combinations of the following single aspects are considered:

- query scheme
- index term set
- term frequency (tf)
- inverse document frequency (idf)
- normalization with respect to document length
- similarity function

Evaluating different *query schemes* (to query search engines for web pages or microblogs) is motivated by the fact that earlier work in web-based MIR has shown an improvement in the accuracy of similarity estimates when adding music-related keywords to the search query (e.g., "music" or "music review") [103, 26, 76]. *Index term set* refers to the list of terms used to filter the microblogs and create the term weight vectors. The number of terms in the index term set corresponds to the dimensionality of the respective feature vectors ($tf \cdot idf$ vectors). The *term frequency* $r_{d,t}$ of a term t in a virtual artist document[36] d estimates the importance t has for document d, hence for the artist under consideration. The *inverse document frequency* w_t estimates the overall importance of term t in the whole corpus and is commonly used to weight the $r_{d,t}$ factor. Performing this calculation for all terms in the used index term set and each virtual artist document results in one $tf \cdot idf$ vector per artist. It is common to subsequently *normalize* the $tf \cdot idf$ vectors with respect to document length. Finally, different *similarity functions* S_{d_i,d_j} to estimate the proximity between the term weight vectors of two virtual artist documents d_i and d_j are examined. Evaluation is carried out using *Mean Average Precision* (MAP) scores on genre labels predicted by various classifiers. This resembles a retrieval task that aims at finding artists of the same genre as the query via similarity.

[36] Virtual artist documents are typically created by concatenating all web pages retrieved for the artist of interest.

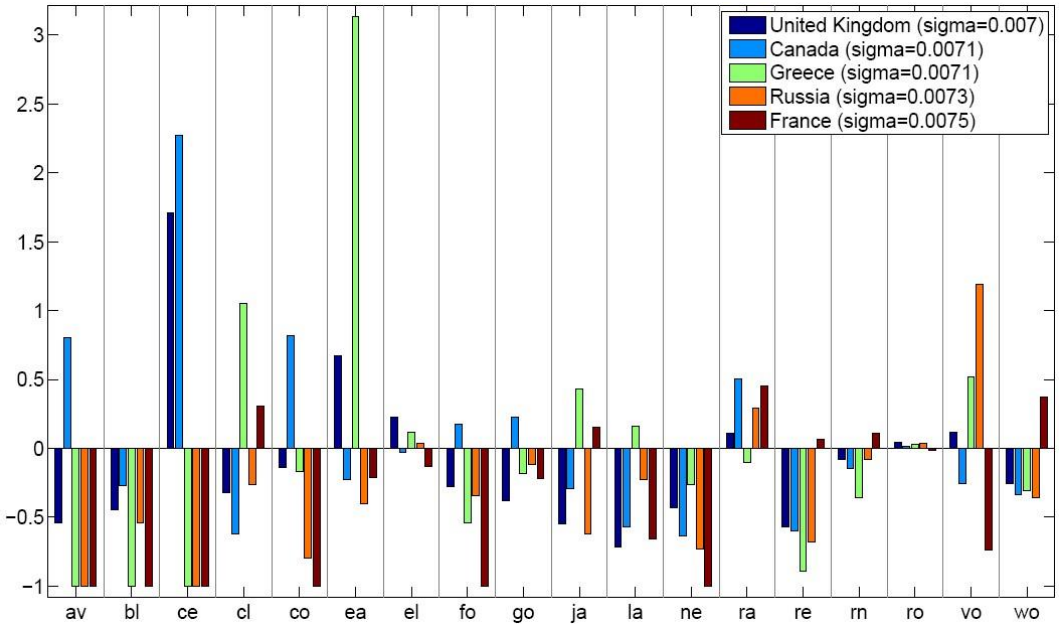


Figure 5 The most "mainstreamy" countries in the world, in terms of music listening behavior.

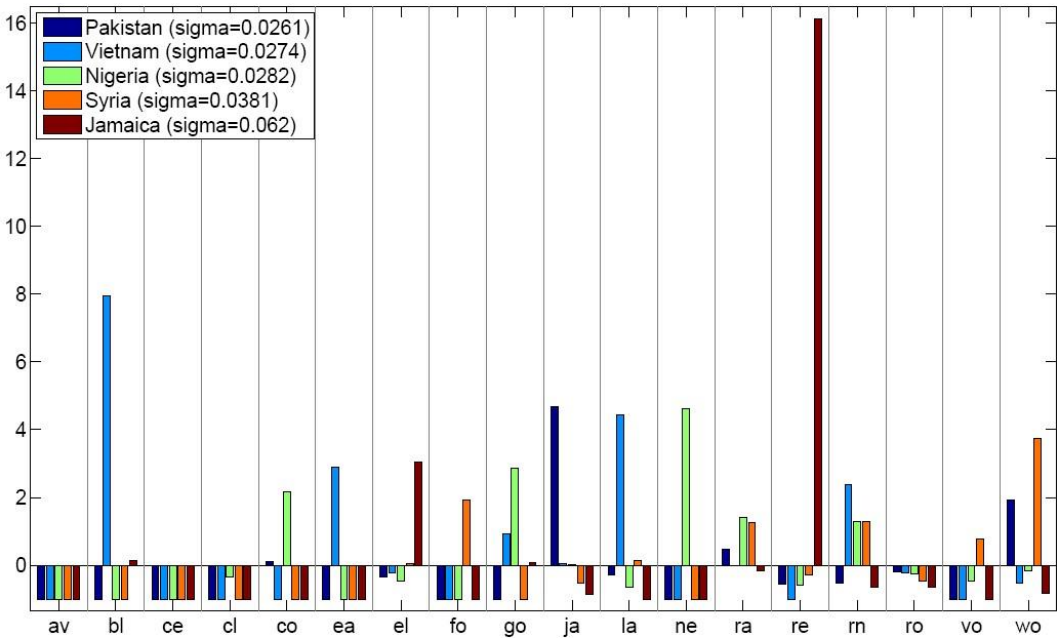


Figure 6 The least "mainstreamy" countries in the world, in terms of music listening behavior.

Findings for web-based similarity: As for the *query scheme* used to determine music-related web page, adding "music" as additional search term to the artist name is preferably over querying only for the artist name, in particular for artists that equal common speech terms (e.g., "Kiss", "Hole", or "Madonna"). Otherwise the retrieved web page set will contain a lot of irrelevant documents.

In Text-IR the entity of interest is usually a text or web document. In MIR, in contrast, we are not interested in investigating individual web pages, but artists or songs. Hence, analyzing different ways to model an artist are considered in the evaluation experiments: (i) aggregate the web pages about each artist to a virtual artist document by concatenation of the individual pages or (ii) computing a statistical summary over the set of pages for each artist (e.g., sum, mean, or maximum of the *tf* or *df* values over the artist's web page set). It was found that the concatenation of individual web pages to *virtual artist documents* outperforms considering each artist web page separately. This holds for the computation of both *tf* and *idf*. Another finding is that *normalization* of each web page, so that each page has the same total weight, shows no improvement in terms of MAP.

As for the *term frequency*, experiments have shown that binary match (i.e., whether or not term t occurs in document d) and simple relative term frequency (i.e., the frequency of term t in document d in relation to the frequency of the most frequent term in d) are not suited for the task at hand. On the other hand, the alternative logarithmic formulation (cf. *tf* component in Equation 17) occurred frequently among the top performing variants.

To model *inverse document frequency*, estimates of term noise (i.e., the noise component in the signal-to-noise ratio), logarithmic *idf* formulations, and entropy measures perform better on average than the other investigated variants.

The overall best performing variants are given by the three term weighting functions in Equations 15, 16, and 17, using *cosine similarity* (Equation 18) as similarity measure between artists. In these equations, N represents the total number of documents in the corpus, $f_{d,t}$ is the number of occurrences of term t in document d, W_d is the document length of d, $\mathcal{T}$ is the set of distinct terms in the corpus, and $\mathcal{T}_{d_1,d_2}$ denotes the set of distinct terms in documents d_1 and d_2. In Equation 16, n_t denotes the noise component in the signal-to-noise ratio for term t, F_t is the total number of occurrences of term t in the corpus, and $\mathcal{D}_t$ is the set of documents in which term t occurs..

$$w_{d,t} = (1 + \log_2 f_{d,t}) \cdot \left(1 - \frac{n_t}{\log_2 N}\right) \tag{15}$$

$$w_{d,t} = (1 + \log_2 f_{d,t}) \cdot \left(\max_{t' \in \mathcal{T}} n_{t'} - n_t\right), \quad n_t = \sum_{d \in \mathcal{D}_t} \left(-\frac{f_{d,t}}{F_t} \log_2 \frac{f_{d,t}}{F_t}\right) \tag{16}$$

$$w_{d,t} = \log_e(1 + f_{d,t}) \cdot \left(1 - \frac{n_t}{\log_2 N}\right) \tag{17}$$

$$sim(d_1, d_2) = \frac{\sum_{t \in \mathcal{T}_{d_1,d_2}} (w_{d_1,t} \cdot w_{d_2,t})}{W_{d_1} \cdot W_{d_2}} \tag{18}$$

As a general finding of the experiments, we conclude that a small change in an algorithmic component (for instance, document frequency computed on web pages vs. on artist level) can have an important impact on the algorithm's performance.

Findings for microblog-based similarity: Experimental results are summarized in Figure 7, which shows the distribution of the MAP values for each algorithmic choice in each categorical aspect under investigation. That is, fixing a particular experimental aspect, for instance, TS_F as term set (cf. top right subfigure), the plot shows a statistical summary

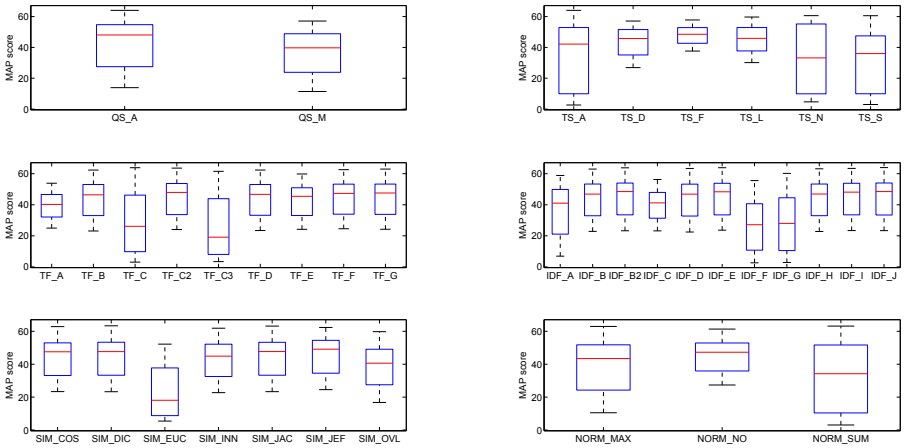


Figure 7 Box plots of MAP scores for each algorithmic choice in the microblog dataset.

(median, minimum, maximum, 0.25- and 0.75-percentile) of all experiments in which term set TS_F is used.

Query Scheme: When dealing with microblogs, it is preferable to use only the artist name (no additional keywords) to query the `Twitter` API and retrieve the tweets relevant to the artist under consideration.

Index Term Sets: Even though using all terms in the corpus yields the highest MAP values, results are by far the most unstable ones (cf. variant TS_A in Figure 7). Slightly modifying a single other aspect can thus cause a significant decline in accuracy when using all terms in the corpus. Given the high computational complexity due to feature spaces of dimensionality greater than one million, employing no particular index term set is not favorable for most retrieval tasks. Good and robust results are achieved using a dictionary of musical genres, musical instruments, and emotions, which was gathered from `Freebase` (cf. variant TS_F).

Term Frequency: A simple binary match *tf* formulation should not be used. The most favorable algorithmic variants are logarithmic formulations and an adapted *Okapi BM25* formulation [59, 58].

Inverse Document Frequency: Among the *idf* formulations, binary match yields the worst results. Also signal estimates and signal-to-noise ratios do not perform much better. Again, logarithmic formulations and the modified *Okapi BM25* formulation yield top results.

Normalization: Performing no normalization for document length performs best, both in terms of accuracy and robustness. This is presumably due to the special characteristics of tweets, which are limited to 140 characters, a limit commonly exhausted by `Twitter` users. Further support for this explanation is given in [94]. Normalization hence does not improve results, just increases computational costs.

Similarity Function: Among the similarity functions under estimation, the *Jeffrey divergence*-based function performs very well, while at the same time maintaining a reasonable stability level. Also the *Jaccard coefficient* performs remarkably well. *Euclidean similarity* performs inferior in all combinations.

Overall, the best performing variants found in the experiments are given by the three term weighting functions in Equations 19, 20, and 21, in combination with the *Jaccard coefficient* similarity function (Equation 22). In these equations, N represents the total number of documents in the corpus, $f_{d,t}$ is the number of occurrences of term t in document d, f_t denominates the total number of documents containing term t, W_d is the document length of d, and $\mathcal{T}_{d_1,d_2}$ denotes the set of distinct terms in documents d_1 and d_2.

$$w_{d,t} = \log_e(1 + f_{d,t}) \cdot \log_e \frac{N - f_t}{f_t} \tag{19}$$

$$w_{d,t} = \log_e(1 + f_{d,t}) \cdot \log \frac{N - f_t + 0.5}{f_t + 0.5} \tag{20}$$

$$w_{d,t} = (1 + \log_e f_{d,t}) \cdot \log_e \frac{N - f_t}{f_t} \tag{21}$$

$$sim(d_1, d_2) = \frac{\sum_{t \in \mathcal{T}_{d_1,d_2}} (w_{d_1,t} \cdot w_{d_2,t})}{W_{d_1}^2 + W_{d_2}^2 - \sum_{t \in \mathcal{T}_{d_1,d_2}} (w_{d_1,t} \cdot w_{d_2,t})} \tag{22}$$

(IV) A system to explore music collections, which combines music content and contextual data:

As already discussed in Section 3.1, music retrieval approaches that are not solely based on music content or on music context data, instead combine the two in an effort to increase user satisfaction, are important for the next generation of intelligent user interfaces to browse potentially huge music collections. Taking a step into this direction, publication [70] [G] proposes a music browsing interface that is capable of dealing with real-world music collections comprising tens of thousands of pieces, unlike most previous graphical browsing interfaces. Dubbed "deepTune", the system employs a hierarchical version of the Self-Organizing Map (SOM) [32] that is trained on rhythm features [53] to cluster the pieces in the collection under consideration.

When navigating through large music collections, user guidance is of particular importance. We hence propose a novel technique to determine prototypical songs that represent each cluster. To this end, we make use of a simple music context feature that is fused with the content-based rhythm features. More precisely, we first approximate the popularity of each artist in the collection via the number of documents related to the artist in `Google`'s web page index. This number is given by the page count estimate $pc(a)$ of artist a's web pages. We then propose a ranking function $r_i(x)$ that takes into account both, audio-based similarity between the pieces and overall familiarity of listeners with the artists in cluster i. For the songs in each cluster i, we thus compute a ranking according to Equation 23, where $r_i^a(x)$ is the ranking given by audio-based similarity and $r_i^w(a)$ is the ranking given by web-based familiarity estimation. We select the highest ranked song to serve as cluster representative.

$$r_i(x) = r_i^a(x) \cdot r_i^w(a) \tag{23}$$

$$r_i^w(a) = norm_{[1,5]}\left(log_{10}\left(pc\left(a\right)\right)\right) \tag{24}$$

$$r_i^a(x) = norm_{[1,2]}\left(\frac{1}{1 + \ln(1 + \|x - m_i\|)}\right) \tag{25}$$

The web-based artist popularity ranking is given in Formula 24, where $norm(\cdot)$ scales the page count estimates to the range $[1, 5]$[37]. The audio-based part of the ranking function

[37] This range was empirically found to yield a good balance between familiarity and similarity of cluster

is given in Equation 25, where x is the feature vector of the music piece under consideration, m_i is the centroid of cluster i in the audio feature space (more precisely, the model vector of map unit i in the trained SOM), $\|\cdot\|$ is the Euclidean distance, and $norm(\cdot)$ is again a normalization function that shifts the range to $[1, 2]$[38]. By fusing these two rankings, we can offer the user anchor points in the visualization that are given by songs both well-known and acoustically similar to the cluster center (cf. Figure 3).

As a final remark, user context-aware information, such as time and location used in publication [66] [D], could be incorporated into visual music browsing interfaces like "deepTune" or its mobile variant "nepDroid" [24]. Doing so is likely to improve user satisfaction and to bring truly multimodal music access systems a step closer to reality.

References

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

2 Jean-Julien Aucouturier, François Pachet, and Mark Sandler. "The Way It Sounds": Timbre Models for Analysis and Retrieval of Music Signals. *IEEE Transactions on Multimedia*, 7(6):1028–1035, December 2005.

3 Claudio Baccigalupo, Enric Plaza, and Justin Donaldson. Uncovering Affinity of Artists to Multiple Genres from Social Behaviour Data. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR)*, Philadelphia, PA, USA, September 2008.

4 Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval – the concepts and technology behind search.* Addison-Wesley, Pearson, Harlow, England, 2nd edition, 2011.

5 Linas Baltrunas, Marius Kaminskas, Bernd Ludwig, Omar Moling, Francesco Ricci, Karl-Heinz Lüke, and Roland Schwaiger. InCarMusic: Context-Aware Music Recommendations in a Car. In *International Conference on Electronic Commerce and Web Technologies (EC-Web)*, Toulouse, France, August–September 2011.

6 Stephan Baumann and Oliver Hummel. Using Cultural Metadata for Artist Recommendation. In *Proceedings of the 3rd International Conference on Web Delivering of Music (WEDELMUSIC)*, Leeds, UK, September 2003.

7 Adam Berenzweig, Beth Logan, Daniel P.W. Ellis, and Brian Whitman. A Large-Scale Evaluation of Acoustic and Subjective Music Similarity Measures. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR)*, Baltimore, MD, USA, October 2003.

8 Eric Brochu, Nando de Freitas, and Kejie Bao. The Sound of an Album Cover: Probabilistic Multimedia and IR. In *Proceedings of the 9th International Workshop on Artificial Intelligence and Statistics*, Key West, FL, USA, January 2003.

9 Michael A. Casey, Remco Veltkamp, Masataka Goto, Marc Leman, Christophe Rhodes, and Malcolm Slaney. Content-Based Music Information Retrieval: Current Directions and Future Challenges. *Proceedings of the IEEE*, 96:668–696, April 2008.

10 Òscar Celma. *Music Recommendation and Discovery – The Long Tail, Long Fail, and Long Play in the Digital Music Space.* Springer, Berlin, Heidelberg, Germany, 2010.

representatives.

[38] This range again was empirically found to yield a good balance between familiarity and similarity of cluster representatives.

11 C.L.A. Clarke, M. Kolla, G.V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and Diversity in Information Retrieval Evaluation. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, Singapore, July 2008.

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

13 Gianluca Demartini, Stefan Siersdorfer, Sergiu Chelaru, and Wolfgang Nejdl. Analyzing Political Trends in the Blogosphere. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM)*, Barcelona, Spain, July 2011.

14 Simon Dooms, Toon De Pessemier, and Luc Martens. A User-centric Evaluation of Recommender Algorithms for an Event Recommendation System. In *Proc. Workshop on Human Decision Making in Recommender Systems (Decisions@RecSys'11) and User-Centric Evaluation of Recommender Systems and Their Interfaces (UCERSTI 2)*, pages 67–73, Chicago, IL, USA, October 2011.

15 J. Stephen Downie. The Scientific Evaluation of Music Information Retrieval Systems: Foundations and Future. *Computer Music Journal*, 28:12–23, June 2004.

16 Daniel P.W. Ellis, Brian Whitman, Adam Berenzweig, and Steve Lawrence. The Quest For Ground Truth in Musical Artist Similarity. In *Proceedings of 3rd International Conference on Music Information Retrieval (ISMIR)*, Paris, France, October 2002.

17 Gijs Geleijnse, Markus Schedl, and Peter Knees. 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)*, Vienna, Austria, September 2007.

18 Asif Ghias, Jonathan Logan, David Chamberlin, and Brian C. Smith. Query by Humming: Musical Information Retrieval in an Audio Database. In *Proceedings of the 3rd ACM International Conference on Multimedia*, San Fancisco, CA, USA, November 1995.

19 Emilia Gómez. *Tonal Description of Music Audio Signals*. PhD thesis, Universitat Pompeu Fabra, Barcelona, Spain, 2006.

20 Masataka Goto and Takayuki Goto. Musicream: New Music Playback Interface for Streaming, Sticking, Sorting, and Recalling Musical Pieces. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR)*, London, UK, September 2005.

21 Sten Govaerts, Nik Corthaut, and Erik Duval. Using Search Engine for Classification: Does It Still Work? In *Proceedings of the IEEE International Symposium on Multimedia (ISM): International Workshop on Advances in Music Information Research (AdMIRe)*, San Diego, CA, USA, December 2009.

22 Sten Govaerts and Erik Duval. A Web-based Approach to Determine the Origin of an Artist. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, Kobe, Japan, October 2009.

23 David Hauger and Markus Schedl. Exploring Geospatial Music Listening Patterns in Microblog Data. In *Proceedings of the 10th International Workshop on Adaptive Multimedia Retrieval (AMR)*, Copenhagen, Denmark, October 2012.

24 Sebastian Huber, Markus Schedl, and Peter Knees. nepDroid: An Intelligent Mobile Music Player. In *Proceedings of the ACM International Conference on Multimedia Retrieval (ICMR)*, Hong Kong, June 2012.

25 Marius Kaminskas and Francesco Ricci. Location-Adapted Music Recommendation Using Tags. In Joseph Konstan, Ricardo Conejo, José Marzo, and Nuria Oliver, editors, *User Modeling, Adaption and Personalization*, volume 6787 of *Lecture Notes in Computer Science*, pages 183–194. Springer Berlin / Heidelberg, 2011.

26 Peter Knees, Elias Pampalk, and Gerhard Widmer. Artist Classification with Web-based Data. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR)*, Barcelona, Spain, October 2004.

27 Peter Knees, Tim Pohle, Markus Schedl, and Gerhard Widmer. 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)*, Amsterdam, the Netherlands, July 2007.

28 Peter Knees and Markus Schedl. Towards Semantic Music Information Extraction from the Web Using Rule Patterns and Supervised Learning. In *Proceedings of the 2nd Workshop on Music Recommendation and Discovery (WOMRAD)*, Chicago, IL, USA, October 2011.

29 Peter Knees, Markus Schedl, Tim Pohle, and Gerhard Widmer. An Innovative Three-Dimensional User Interface for Exploring Music Collections Enriched with Meta-Information from the Web. In *Proceedings of the 14th ACM International Conference on Multimedia*, Santa Barbara, California, USA, October 2006.

30 Peter Knees, Markus Schedl, Tim Pohle, and Gerhard Widmer. Exploring Music Collections in Virtual Landscapes. *IEEE MultiMedia*, 14(3):46–54, July–September 2007.

31 Noam Koenigstein, Yuval Shavitt, and Tomer Tankel. Spotting Out Emerging Artists Using Geo-Aware Analysis of P2P Query Strings. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 937–945, Las Vegas, NV, USA, August 2008.

32 Teuvo Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Germany, 3rd edition, 2001.

33 Paul Lamere. Social Tagging and Music Information Retrieval. *Journal of New Music Research: Special Issue: From Genres to Tags – Music Information Retrieval in the Age of Social Tagging*, 37(2):101–114, 2008.

34 Cyril Laurier, Jens Grivolla, and Perfecto Herrera. Multimodal Music Mood Classification using Audio and Lyrics. In *Proceedings of the International Conference on Machine Learning and Applications (ICMLA)*, San Diego, CA, USA, December 2008.

35 E. Law, L. von Ahn, R. Dannenberg, and M. Crawford. Tagatune: A Game for Music and Sound Annotation. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 2007.

36 Edith Law and Luis von Ahn. Input-Agreement: A New Mechanism for Collecting Data Using Human Computation Games. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems (CHI)*, Boston, MA, USA, April 2009.

37 Mark Levy and Mark Sandler. A semantic space for music derived from social tags. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 2007.

38 Cynthia C.S. Liem, Meinard Müller, Douglas Eck, George Tzanetakis, and Alan Hanjalic. The Need for Music Information Retrieval with User-centered and Multimodal Strategies. In *Proceedings of the 1st International ACM Workshop on Music Information Retrieval with User-centered and Multimodal Strategies*, Scottsdale, AZ, USA, November 2011.

39 Beth Logan. Mel Frequency Cepstral Coefficients for Music Modeling. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, Plymouth, MA, USA, October 2000.

40 Beth Logan, Daniel P.W. Ellis, and Adam Berenzweig. Toward Evaluation Techniques for Music Similarity. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR): Workshop on the Evaluation of Music Information Retrieval Systems*, Toronto, Canada, July–August 2003.

41 Dominik Lübbers and Matthias Jarke. Adaptive Multimodal Exploration of Music Collections. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, Kobe, Japan, October 2009.

42 Michael I. Mandel and Daniel P.W. Ellis. A Web-based Game for Collecting Music Metadata. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 2007.

43 Brian McFee, Thierry Bertin-Mahieux, Dan Ellis, and Gert Lanckriet. The Million Song Dataset Challenge. In *Proceedings of the 21st International World Wide Web Conference (WWW): 4th International Workshop on Advances in Music Information Research (AdMIRe)*, Lyon, France, April 2012.

44 Brian McFee and Gert Lanckriet. Hypergraph Models of Playlist Dialects. In *Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR)*, Porto, Portugal, October 2012.

45 Hirokazu Kameoka Nobutaka Ono, Kenichi Miyamoto and Shigeki Sagayama. A real-time equalizer of harmonic and percussive. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR)*, Philadelphia, PA, USA, September 2008.

46 Nicola Orio, Cynthia C.S. Liem, Geoffroy Peeters, and Markus Schedl. MusiClef: Multimodal Music Tagging Task. In *Proceedings of the 3rd Conference on Multilingual and Multimodal Information Access Evaluation (CLEF)*, Rome, Italy, September 2012.

47 Nicola Orio, David Rizo, Riccardo Miotto, Nicola Montecchio, Markus Schedl, and Olivier Lartillot. MusiCLEF: A Benchmark Activity in Multimodal Music Information Retrieval. In *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, Miami, FL, USA, October 2011.

48 François Pachet, Gert Westerman, and Damien Laigre. Musical Data Mining for Electronic Music Distribution. In *Proceedings of the 1st International Conference on Web Delivering of Music (WEDELMUSIC)*, Florence, Italy, November 2001.

49 Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank Citation Ranking: Bringing Order to the Web. In *Proceedings of the Annual Meeting of the American Society for Information Science (ASIS)*, pages 161–172, January 1998.

50 Elias Pampalk, Arthur Flexer, and Gerhard Widmer. 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)*, Vienna, Austria, September 2005.

51 Elias Pampalk and Masataka Goto. MusicRainbow: A New User Interface to Discover Artists Using Audio-based Similarity and Web-based Labeling. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR)*, Victoria, Canada, October 2006.

52 Elias Pampalk and Masataka Goto. MusicSun: A New Approach to Artist Recommendation. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 2007.

53 Elias Pampalk, Andreas Rauber, and Dieter Merkl. Content-based Organization and Visualization of Music Archives. In *Proceedings of the 10th ACM International Conference on Multimedia*, pages 570–579, Juan les Pins, France, December 2002.

54 Tim Pohle, Peter Knees, Markus Schedl, Elias Pampalk, and Gerhard Widmer. "Reinventing the Wheel": A Novel Approach to Music Player Interfaces. *IEEE Transactions on Multimedia*, 9:567–575, 2007.

55 Tim Pohle, Dominik Schnitzer, Markus Schedl, Peter Knees, and Gerhard Widmer. On Rhythm and General Music Similarity. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, Kobe, Japan, October 2009.

56 Pearl Pu, Li Chen, and Rong Hu. A User-Centric Evaluation Framework for Recommender Systems. In *Proceedings of the ACM Recommender Systems (RecSys)*, pages 157–164, Chicago, IL, USA, October 2011.

57 M. Radovanović, A. Nanopoulos, and M. Ivanović. Hubs in Space: Popular Nearest Neighbors in High-dimensional Data. *The Journal of Machine Learning Research*, pages 2487–2531, 2010.

58 S.E. Robertson, S. Walker, and M. Beaulieu. Okapi at TREC-7: Automatic Ad Hoc, Filtering, VLC and Interactive Track. In *Proceedings of the 7th Text REtrieval Conference (TREC-7)*, 1999.

59 S.E. Robertson, S. Walker, and M.M. Hancock-Beaulieu. Large Test Collection Experiments on an Operational, Interactive System: Okapi at TREC. *Information Processing & Management*, 31, 1995.

60 Markus Schedl. *Automatically Extracting, Analyzing, and Visualizing Information on Music Artists from the World Wide Web*. PhD thesis, Johannes Kepler University, Linz, Austria, 2008.

61 Markus Schedl. On the Use of Microblogging Posts for Similarity Estimation and Artist Labeling. In *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR)*, Utrecht, the Netherlands, August 2010.

62 Markus Schedl. Analyzing the Potential of Microblogs for Spatio-Temporal Popularity Estimation of Music Artists. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI): International Workshop on Social Web Mining*, Barcelona, Spain, July 2011.

63 Markus Schedl. *Music Data Mining*, chapter Web-Based and Community-Based Music Information Extraction. CRC Press/Chapman Hall, 2011.

64 Markus Schedl. #nowplaying Madonna: A Large-Scale Evaluation on Estimating Similarities Between Music Artists and Between Movies from Microblogs. *Information Retrieval*, 15:183–217, June 2012.

65 Markus Schedl. *Social Media Retrieval*, chapter Exploiting Social Media for Music Information Retrieval. Springer, December 2012.

66 Markus Schedl. Leveraging Microblogs for Spatiotemporal Music Information Retrieval. In *Proceedings of the 35th European Conference on Information Retrieval (ECIR)*, Moscow, Russia, March 2013.

67 Markus Schedl, Arthur Flexer, and Julián Urbano. The Neglected User in Music Information Retrieval Research. *Journal of Intelligent Information Systems*, 2013.

68 Markus Schedl and David Hauger. Mining Microblogs to Infer Music Artist Similarity and Cultural Listening Patterns. In *Proceedings of the 21st International World Wide Web Conference (WWW): 4th International Workshop on Advances in Music Information Research (AdMIRe)*, Lyon, France, April 2012.

69 Markus Schedl, David Hauger, and Dominik Schnitzer. A Model for Serendipitous Music Retrieval. In *Proceedings of the 16th International Conference on Intelligent User Interfaces (IUI): 2nd International Workshop on Context-awareness in Retrieval and Recommendation (CaRR)*, Lisbon, Portugal, February 2012.

70 Markus Schedl, Christian Höglinger, and Peter Knees. Large-Scale Music Exploration in Hierarchically Organized Landscapes Using Prototypicality Information. In *Proceedings of the ACM International Conference on Multimedia Retrieval (ICMR)*, Trento, Italy, April 2011.

71 Markus Schedl and Peter Knees. Investigating Different Term Weighting Functions for Browsing Artist-Related Web Pages by Means of Term Co-Occurrences. In *Proceedings of the 2nd International Workshop on Learning the Semantics of Audio Signals (LSAS)*, Paris, France, June 2008.

72 Markus Schedl and Peter Knees. Context-based Music Similarity Estimation. In *Proceedings of the 3rd International Workshop on Learning the Semantics of Audio Signals (LSAS)*, Graz, Austria, December 2009.

73 Markus Schedl and Peter Knees. Personalization in Multimodal Music Retrieval. In *Proceedings of the 9th Workshop on Adaptive Multimedia Retrieval (AMR)*, Barcelona, Spain, July 2011.

74 Markus Schedl, Peter Knees, and Sebastian Böck. Investigating the Similarity Space of Music Artists on the Micro-Blogosphere. In *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, Miami, FL, USA, October 2011.

75 Markus Schedl, Peter Knees, Tim Pohle, and Gerhard Widmer. Towards Automatic Retrieval of Album Covers. In *Proceedings of the 28th European Conference on Information Retrieval (ECIR)*, London, UK, April 2006.

76 Markus Schedl, Peter Knees, and Gerhard Widmer. A Web-Based Approach to Assessing Artist Similarity using Co-Occurrences. In *Proceedings of the 4th International Workshop on Content-Based Multimedia Indexing (CBMI)*, Riga, Latvia, June 2005.

77 Markus Schedl, Peter Knees, and Gerhard Widmer. 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)*, Bangalore, India, December 2006.

78 Markus Schedl, Cynthia C.S. Liem, Geoffroy Peeters, and Nicola Orio. A Professionally Annotated and Enriched Multimodal Data Set on Popular Music. In *Proceedings of the 4th ACM Multimedia Systems Conference (MMSys)*, Oslo, Norway, February–March 2013.

79 Markus Schedl and Tim Pohle. Enlightening the Sun: A User Interface to Explore Music Artists via Multimedia Content. *Multimedia Tools and Applications: Special Issue on Semantic and Digital Media Technologies*, 49(1):101–118, August 2010.

80 Markus Schedl, Tim Pohle, Peter Knees, and Gerhard Widmer. Exploring the Music Similarity Space on the Web. *ACM Transactions on Information Systems*, 29(3), July 2011.

81 Markus Schedl, Tim Pohle, Noam Koenigstein, and Peter Knees. What's Hot? Estimating Country-Specific Artist Popularity. In *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR)*, Utrecht, the Netherlands, August 2010.

82 Markus Schedl, Cornelia Schiketanz, and Klaus Seyerlehner. Country of Origin Determination via Web Mining Techniques. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, Singapore, July 2010.

83 Markus Schedl and Dominik Schnitzer. Hybrid Retrieval Approaches to Geospatial Music Recommendation. In *Proceedings of the 35th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, Dublin, Ireland, July–August 2013.

84 Markus Schedl, Klaus Seyerlehner, Dominik Schnitzer, Gerhard Widmer, and Cornelia Schiketanz. Three Web-based Heuristics to Determine a Person's or Institution's Country of Origin. In *Proceedings of the 33th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, Geneva, Switzerland, July 2010.

85 Markus Schedl, Sebastian Stober, Emilia Gómez, Nicola Orio, and Cynthia C.S. Liem. User-Aware Music Retrieval. In Meinard Müller, Masataka Goto, and Markus Schedl, editors, *Multimodal Music Processing*, volume 3 of *Dagstuhl Follow-Ups*, pages 135–156. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2012.

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

87 Markus Schedl, Gerhard Widmer, Peter Knees, and Tim Pohle. A Music Information System Automatically Generated via Web Content Mining Techniques. *Information Processing & Management*, 47, 2011.

88 Markus Schedl, Gerhard Widmer, Tim Pohle, and Klaus Seyerlehner. Web-based Detection of Music Band Members and Line-Up. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 2007.

89 Eric D. Scheirer. Tempo and Beat Analysis of Acoustic Musical Signals. *Journal of Acoustical Society of America*, 103(1):588–601, January 1998.

90 Dominik Schnitzer, Arthur Flexer, Markus Schedl, and Gerhard Widmer. Local and global scaling reduce hubs in space. *Journal of Machine Learning Research*, 13:2871–2902, October 2012.

91 Klaus Seyerlehner, Reinhard Sonnleitner, Markus Schedl, David Hauger, and Bogdan Ionescu. From Improved Auto-taggers to Improved Music Similarity Measures. In *Proceedings of the 10th International Workshop on Adaptive Multimedia Retrieval (AMR)*, Copenhagen, Denmark, October 2012.

92 Klaus Seyerlehner, Gerhard Widmer, Markus Schedl, and Peter Knees. Automatic Music Tag Classification based on Block-Level Features. In *Proceedings of the 7th Sound and Music Computing Conference (SMC)*, Barcelona, Spain, July 2010.

93 Yuval Shavitt and Udi Weinsberg. Songs Clustering Using Peer-to-Peer Co-occurrences. In *Proceedings of the IEEE International Symposium on Multimedia (ISM): International Workshop on Advances in Music Information Research (AdMIRe)*, San Diego, CA, USA, December 2009.

94 Bharath Sriram, Dave Fuhry, Engin Demir, Hakan Ferhatosmanoglu, and Murat Demirbas. Short Text Classification in Twitter to Improve Information Filtering. In *Proceedings of the 33th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, Geneva, Switzerland, July 2010.

95 Sebastian Stober. *Adaptive Methods for User-Centered Organization of Music Collections*. PhD thesis, Otto-von-Guericke-University, Magdeburg, Germany, November 2011. published by Dr. Hut Verlag, ISBN 978-3-8439-0229-8.

96 Gabor Szabo and Bernardo A. Huberman. Predicting the Popularity of Online Content. *Communications of the ACM*, 53(8):80–88, August 2010.

97 D. Turnbull, R. Liu, L. Barrington, and G. Lanckriet. A Game-based Approach for Collecting Semantic Annotations of Music. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, September 2007.

98 George Tzanetakis and Perry Cook. Musical Genre Classification of Audio Signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, 2002.

99 Julián Urbano and Markus Schedl. Towards Minimal Test Collections for Evaluation of Audio Music Similarity and Retrieval. In *Proceedings of the 21st International World Wide Web Conference (WWW): 4th International Workshop on Advances in Music Information Research (AdMIRe)*, Lyon, France, April 2012.

100 Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.

101 Luis von Ahn and Laura Dabbish. Labeling Images with a Computer Game. In *Proceedings of the 22nd International Conference on Human Factors in Computing Systems (CHI)*, Vienna, Austria, April 2004.

102 Brain Whitman, Gary Flake, and Steve Lawrence. Artist Detection in Music with Minnowmatch. In *Proceedings of the IEEE Workshop on Neural Networks for Signal Processing*, Falmouth, MA, USA, September 2001.

103 Brian Whitman and Steve Lawrence. Inferring Descriptions and Similarity for Music from Community Metadata. In *Proceedings of the 2002 International Computer Music Conference (ICMC)*, Göteborg, Sweden, September 2002.

104 Daniel Wolff and Tillman Weyde. Adapting Metrics for Music Similarity using Comparative Ratings. In *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, Miami, FL, USA, October 2011.

105 Mark Zadel and Ichiro Fujinaga. Web Services for Music Information Retrieval. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR)*, Barcelona, Spain, October 2004.

106 Eva Zangerle, Wolfgang Gassler, and Günther Specht. Exploiting Twitter's Collective Knowledge for Music Recommendations. In *Proceedings of the 21st International World Wide Web Conference (WWW): Making Sense of Microposts (#MSM)*, Lyon, France, April 2012.

107 Justin Zobel and Alistair Moffat. Inverted Files for Text Search Engines. *ACM Computing Surveys*, 38:1–56, 2006.

Part II

Core Publications

Institut für Computational Perception
Johannes Kepler Universität, Linz, Austria

Markus Schedl, Arthur Flexer, Julián Urbano

The Neglected User in Music Information Retrieval Research

Journal of Intelligent Information Systems, 2013

Journal of Intelligent Information Systems manuscript No.
(will be inserted by the editor)

The Neglected User in Music Information Retrieval Research

Markus Schedl · Arthur Flexer · Julián Urbano

Abstract Personalization and context-awareness are highly important topics in research on Intelligent Information Systems. In the fields of Music Information Retrieval (MIR) and Music Recommendation in particular, user-centric algorithms should ideally provide music that perfectly fits each individual listener in each imaginable situation and for each of her information or entertainment needs. Even though preliminary steps towards such systems have recently been presented at the "International Society for Music Information Retrieval Conference" (ISMIR) and at similar venues, this vision is still far away from becoming a reality. In this article, we investigate and discuss literature on the topic of user-centric music retrieval and reflect on why the breakthrough in this field has not been achieved yet. Given the different expertises of the authors, we shed light on why this topic is a particularly challenging one, taking *computer science* and *psychology* points of view. Whereas the computer science aspect centers on the problems of user modeling, machine learning, and evaluation, the psychological discussion is mainly concerned with proper experimental design and interpretation of the results of an experiment. We further present our ideas on aspects crucial to consider when elaborating user-aware music retrieval systems.

Keywords user-centric music retrieval · experimental design · evaluation · interpretation

This research is supported by the Austrian Science Fund (FWF): P22856 and P24095, by the Spanish Government: HAR2011-27540, and by the European Commission, FP7 (Seventh Framework Programme), ICT-2011.1.5 Networked Media and Search Systems, grant agreement no. 287711. The research leading to these results has received funding from the European Union Seventh Framework Programme FP7 / 2007-2013 through the PHENICX project under grant agreement no. 601166.

Markus Schedl
Department of Computational Perception
Johannes Kepler University, Linz, Austria
E-mail: markus.schedl@jku.at

Arthur Flexer
Austrian Research Institute for Artificial Intelligence, Vienna
E-mail: arthur.flexer@ofai.at

Julián Urbano
Department of Computer Science
University Carlos III of Madrid, Spain
E-mail: jurbano@inf.uc3m.es

1 Why care about the user?

In our discussion of the importance and the challenges of development and evaluation in Music Information Retrieval (MIR) we distinguish between *systems-based* and *user-centric* MIR. We define systems-based MIR as all research concerned with laboratory experiments existing solely in a computer, e.g. evaluation of algorithms on digital databases. In contrast, user-centric MIR always involves human subjects and their interaction with MIR systems.

Systems-based MIR has traditionally focused on computational models to describe universal aspects of human music perception, for instance, via elaborating musical feature extractors or similarity measures. Doing so, the existence of an objective "ground truth" is assumed, against which corresponding music retrieval algorithms (e.g., playlist generators or music recommendation systems) are evaluated. To give a common example, music retrieval approaches have been evaluated via genre classification experiments for years. Although it was shown already in 2003 that musical genre is an ill-defined concept [2], genre information still serves as a proxy to vaguely assess music similarity and retrieval approaches in systems-based MIR.

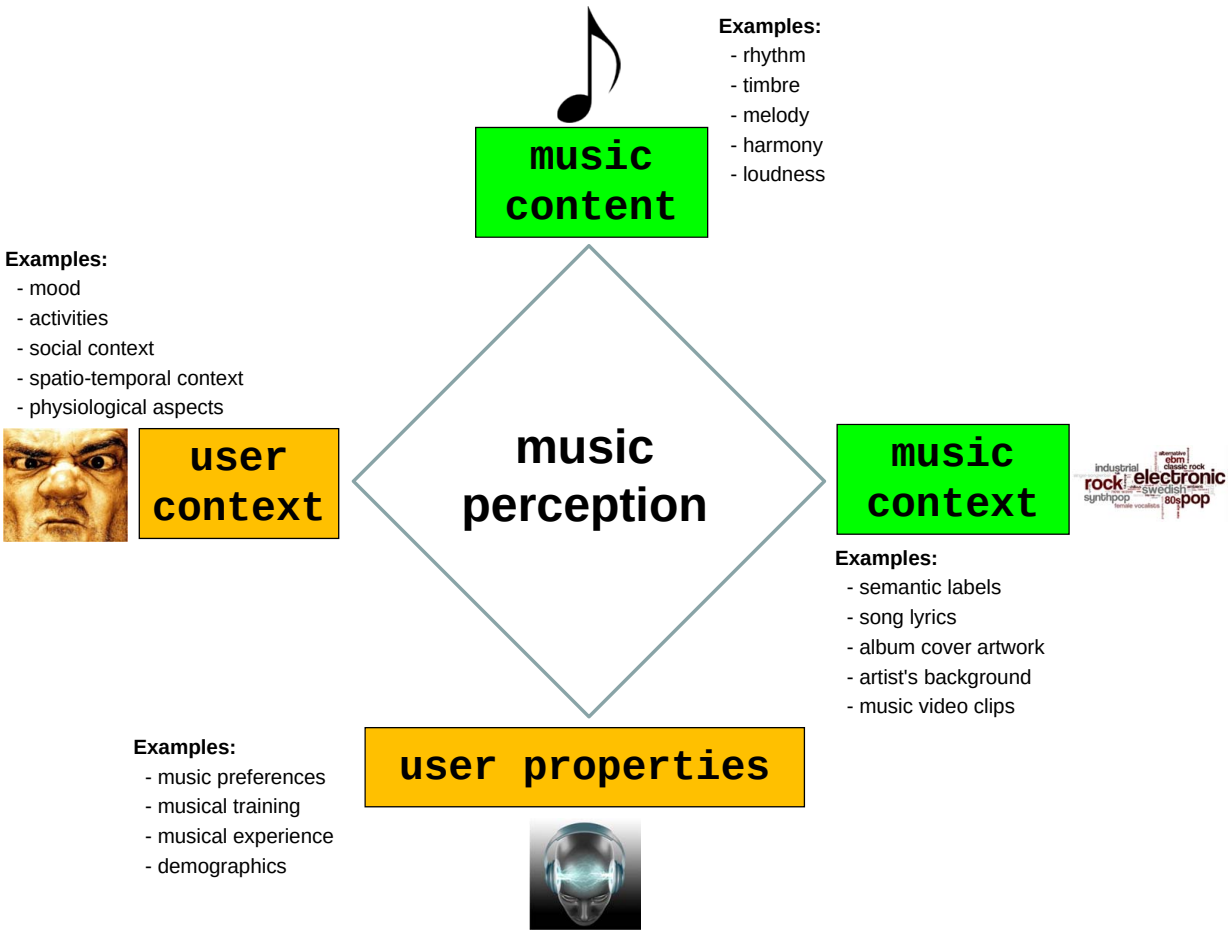


Fig. 1 Factors that influence human music perception.

On the way towards user-centric MIR, the coarse and ambiguous concept of genre should either be treated in a personalized way or replaced by the concept of similarity. When humans are asked to judge the similarity between two pieces of music, however, certain other challenges need to be faced. Common evaluation strategies typically do not take into account the musical expertise and taste of the users. A clear definition of "similarity" is often missing too. It might hence easily occur that two users apply a very different, individual notion of similarity when assessing the output of music retrieval systems. While a first person may experience two songs as rather dissimilar due to very different lyrics, a second one may feel a much higher resemblance of the very same songs because of a similar instrumentation. Similarly, a fan of Heavy Metal music might perceive a Viking Metal track as dissimilar to a Death Metal piece, while for the majority of people the two will sound alike. Scientific evidence for this subjective perception of musical similarity can be found, for instance, in [38] in which a new kind of "game with a purpose" is proposed. Named "TagATune", the aim of this 2-player-game is to decide if two pieces the players listen to simultaneously are the same or not. To this end, they are allowed to exchange free-form labels, tags, or other text. In a bonus round, players are presented three songs, one seed and two target songs. They now have to decide, which of the two targets is more similar to the seed. Based on an analysis of the data collected in this bonus round, [60] show that there are many tuples on which users do not agree. A more decent investigation of perceptual human similarity is performed in [46], where Novello et al. analyze concordance of relative human similarity judgments gathered by an experiment similar to the TagATune bonus rounds. The experiment included 36 participants who had to judge the same set of 102 triads each. Although the authors report statistically significant concordance values for 95% of the triads (measured via Kendall's coefficient of rank correlation), only about half of the triads show a correlation value higher than 0.5, which is frequently taken as indicator of a moderate correlation.

Analyzing how users organize their music collections and which methods they apply to browse them or seek

for particular music has not been of major interest in the MIR community, although this topic is certainly related to user-centric MIR. Work in the corresponding research area is carried out to a large extent by Sally Jo Cunningham and colleagues, who dubbed it "music information behavior". For instance, [17] reports on a study performed via interviews and on-site observations, aiming at investigating how people organize their music collections. Their findings include that (i) a person's physical music collection is frequently divided into "active items" and "archival items", (ii) albums are frequently sorted according to date of purchase, release date, artist in alphabetic order, genre, country of origin, most favorite to least favorite, or recency of being played, and (iii) music is frequently organized according to the intended use, for instance, a particular event or occasion.

Looking into user behavior when it comes to constructing playlists, Cunningham et al. carried out in [15] a qualitative study based on user questionnaires and postings of related web sites. They found that users frequently start creating a playlist by browsing through their music collections in a linear manner or by considering their recent favorite songs. Cunningham et al. further criticize that most music retrieval systems are missing a function to explicitly exclude songs with a particular attribute (e.g., music of a particular genre or by a certain artist). Given the results of another study [16], which aimed at assessing which songs are the most hated ones, such a function would be vital, though.

More recent work looks into music listening and organization behavior via online surveys [31] or tackle specific user groups, for instance, homeless people in North America [59]. The former study found that the most important attributes used to organize music are artist, album, and genre. When it comes to creating playlists, also mood plays an important role. Furthermore, the study showed a strong correlation between user activities (in particular, high attention and low attention activities) and aspects such as importance, familiarity, and mood of songs, as well as willingness to interact with the player. The latter study [59] investigates the reasons for listening to music, among homeless people. It reveals that calming down, help to get through difficult times, and just to relieve boredom are the most important driving factors why homeless young people in Vancouver, Canada, listen to music.

The above examples and analyses illustrate that there are many aspects that influence what a human perceives as similar in a musical context. According to [54], these aspects can be grouped into three different categories: *music content*, *music context*, and *user context*. Here we extend our previous categorization [54] by a fourth set of aspects, the *user properties*. Examples for each category are given in Figure 1. Broadly speaking, *music content* refers to all aspects that are encoded in and can be inferred from the audio signal, while *music context* includes factors that cannot be extracted directly from the audio, but are nevertheless related to the music item, artist, or performer. For instance, rhythmic structure, melody, and timbre features belong to the former category, whereas information about the artist's cultural or political background, collaborative semantic labels, and album cover artwork fall into the latter category. While *user context* aspects represent dynamic and frequently changing factors, such as the user's current social context or activity, *user properties* refer to constant or only slowly changing features of the user, such as her music taste or skills in playing instruments. The incorporation of user context and user properties into our model of music perception is also justified by the analysis reported in [25] about how people communicate using music. In particular, Hargreaves et al. highlight the importance of "non-music context" both for communicating through music and for the listeners' perception of music. The authors give some examples of such context categories and particular aspects: social and cultural context (political and national context), everyday situations (work, leisure, consumer, entertainment), presence/absence of others (live, audience, recorder).

It is exactly this multifaceted and individual way of music perception that has largely been neglected so far when elaborating and evaluating music retrieval approaches, but should be given more attention, in particular considering the trend towards personalized and context-aware systems [41, 54].

A *personalized system* is one that incorporates information about the user into its data processing part (e.g., a particular user taste for a movie genre). A *context-aware system*, in contrast, takes into account dynamic aspects of the user context when processing the data (e.g., location and time where/when a user issues a query). Although the border between personalization and context-awareness may appear fuzzy from this definition, in summary, personalization usually refers to the incorporation of more static, general user preferences, whereas context-awareness refers to the fact that frequently changing aspects of the user's environmental, psychological, and physiological context are considered. Given our categorization of aspects influencing music perception (Figure 1), generally speaking, personalization draws on factors in the category *user properties*, whereas context-aware models encompass aspects of the *user context*.

In discussing these aspects of user-centric MIR, we will take both a computer science and psychological point of view. The computer science aspect is mainly concerned with the algorithmic and computational challenges of modeling individual or groups of users in MIR. Our psychological approach concentrates on proper experimental design and interpretation of results. Of course we are aware that psychology is a much broader field and that music psychology in particular tries to explain both musical behavior and musical experience as a whole with psychological methods. Discussion of this broader field of common interests is beyond the scope of this paper and we like to point interested readers to a joint presentation of an eminent MIR researcher and a psychologist elaborating on the sometimes complicated dialog of the two disciplines at last year's major conference in the MIR field (ISMIR 2012[1]) [1].

The remainder of this paper is organized as follows. Section 2 reviews approaches that, in one way or the other, take the user into account when building music retrieval systems. We also discuss here the role of the user in communities other than MIR and analyze what the MIR community can learn from others. Evaluation strategies for investigating user-centric MIR are discussed in Section 3. In Section 4, we eventually summarize important factors when creating and evaluating user-aware music retrieval systems.

2 How to model the user?

Existing personalized and user-aware systems typically model the user in a very simplistic way. For instance, it is common in *collaborative filtering* approaches [52, 42] to build user profiles only from information about a user u expressing an interest in item i. As an indicator of interest may serve, for example, a click on a particular item, a purchasing transaction, or in MIR the act of listening to a certain music piece. Such indications, in their simplest form, are stored in a binary matrix where element $r(u, i)$ denotes the presence or absence of a connection between user u and item i. In common recommendation systems, a more fine-grained scale for modeling the user interest in an item is typically employed – users frequently rate items according to a Likert-type scale, e.g., by assigning one to five stars to it. Matrix factorization techniques are subsequently applied to recommend novel items [37].

In the following, we first analyze the role of the user in literature about MIR (Section 2.1). We then look at how other communities, in particular the Recommendation Systems community, address the user and what the MIR community can learn from these (Section 2.2).

2.1 What about the user in MIR?

Taking a closer look at literature about context-aware retrieval and recommendation in the music domain, we can see that approaches differ considerably in terms of how the user context is defined, gathered, and incorporated. A summary and categorization of corresponding scientific works can be found in Table 1. The majority of approaches rely solely on one or a few aspects (temporal features in [12], listening history and weather conditions in [40], for instance), whereas more comprehensive user models are rare in MIR. One of the few exceptions is Cunningham et al.'s study [14] that investigates if and how various factors relate to music taste (e.g., human movement, emotional status, and external factors such as temperature and lightning conditions). Based on the findings, the authors present a fuzzy logic model to create playlists.

There further exists some work that assumes a mobile music consumption scenario. The corresponding systems frequently aim at matching music with the current pace of a walker or jogger, e.g. [45,6]. Such systems typically try to match the user's heartbeat with the music played [43]. However, almost all proposed systems require additional hardware for context logging, e.g. [21, 19,14].

In [32] a system that matches tags describing a particular place with tags describing music is presented. Employing text-based similarity measures between the multimodal sets of tags, Kaminskas and Ricci propose their system for location-based music recommendation. Baltrunas et al. [5] suggest a context-aware music recommender system for music consumption while driving. Although the authors take into account eight different contextual factors (e.g., driving style, mood, road type, weather, traffic conditions), their application scenario is quite restricted and their system relies on explicit human feedback, which is burdensome.

Zhang et al. present *CompositeMap* [63], a model that takes into account similarity aspects derived from music content as well as social factors. The authors propose a multimodal music similarity measure and show its applicability to the task of music retrieval. They also allow a simple kind of personalization of this model by letting the user weight the individual music dimensions on which similarity is estimated. However, they do neither take the user context into consideration, nor do they try to learn a user's preferences.

[1] `http://ismir2012.net`

Table 1 Categorization of literature about music retrieval including user aspects.

Features	music content	(Zhang et al., 2009) [63], (Knees and Widmer, 2007) [34], (Nürnberger and Detyniecki, 2003) [47]
	music context	(Kaminskas and Ricci, 2011) [32], (Zhang et al., 2009) [63], (Pohle et al., 2007) [48], (Knees and Widmer, 2007) [34]
	user-centric	(Cebrián et al., 2010) [12] – few features, (Lee and Lee, 2007) [40] – few features, (Cunningham et al., 2008) [14] – many features, (Xue et al., 2009) [62] – features at different levels
Personalization	relevance feedback	(Knees and Widmer, 2007) [34]
	user-adjustable weights	(Zhang et al., 2009) [63], (Pohle et al., 2007) [48], (Nürnberger and Detyniecki, 2003) [47]
Context-Aware	restricted to "sports"	(Moens et al., 2010) [45], (Liu et al., 2009) [43], (Biehl et al., 2006) [6], (Elliott and Tomlinson, 2006) [21], (Dornbush et al., 2007) [19], (Cunningham et al., 2008) [14]
	restricted to "driving a car"	(Baltrunas et al., 2011) [5]
	restricted to "places of interest"	(Kaminskas and Ricci, 2011) [32]
Evaluation	no user involvement reported	(Cebrián et al., 2010) [12], (Pohle et al., 2007) [48], (Nürnberger and Detyniecki, 2003) [47]
	precompiled user-generated data sets	(Xue et al., 2009) [62], (Knees et al., 2007) [33], (Lee and Lee, 2007) [40]
	user response to single question	(Kaminskas and Ricci, 2011) [32], (Liu et al., 2009) [43], (Moens et al., 2010) [45], (Biehl et al., 2006) [6]
	multifaceted questionnaire	(Bogdanov and Herrera, 2011) [7], (Firan et al., 2007) [22]

In [48] Pohle et al. present preliminary steps towards a simple personalized music retrieval system. Based on a clustering of community-based tags extracted from *Last.fm*, a small number of musical concepts are derived using *Non-Negative Matrix Factorization* (NMF) [39,61]. Each music artist or band is then described by a "concept vector". A user interface allows for adjusting the weights of the individual concepts, based on which artists that best match the resulting distribution of the concepts are recommended to the user. Zhang et al. propose in [63] a very similar kind of personalization strategy via user-adjusted weights.

Knees and Widmer present in [34] an approach that incorporates *relevance feedback* [51] into a text-based music search engine [33] to adapt the retrieval process to user preferences. The search engine proposed by Knees et al. builds a model from music content features (MFCCs) and music context features (term vector representations of artist-related Web pages). To this end, a weight is computed for each (term, music item)-pair, based on the term vectors. These weights are then smoothed, taking into account the closest neighbors according to the content-based similarity measure (Kullback-Leibler divergence on Gaussian Mixture Models of the MFCCs). To retrieve music via natural language queries, each textual query issued to the system is expanded via a *Google* search, resulting again in a term weight vector. This query vector is subsequently compared to the smoothed weight vectors describing the music pieces, and those with smallest distance to the query vector are returned.

Nürnberger and Detyniecki present in [47] a variant of the *Self-Organizing Map* (SOM) [36] that is based on a model that adapts to *user feedback*. To this end, the user can move data items on the SOM. This information is fed back into the SOM's codebook, and the mapping is adapted accordingly.

In [62] Xue et al. present a *collaborative personalized search model* that alleviates the problems of *data sparseness* and *cold-start for new users* by combining information on different levels (individual, interest group, and global). Although not explicitly targeted at music retrieval, the idea of integrating data about the user, his peer group, and global data to build a social retrieval model might be worth considering for MIR purposes.

The problem with the vast majority of approaches presented so far is that evaluation is still carried out without sufficient user involvement. For instance, [12, 48,47] seemingly do not perform any kind of evaluation involving real users, or at least do not report it. Some approaches are evaluated on user-generated data, but do not request feedback from real users during the evaluation experiments. For example, [33] makes use of collaborative tags stored in a database to evaluate the proposed music search engine. Similarly, [40] relies on data sets of listening histories and weather conditions, and [62] uses a corpus of Web search data. Even if real users are questioned during evaluation, their individual properties (such as taste, expertise, or familiarity with the music items under investigation) are regularly neglected in evaluation experiments. In these cases, evaluation is typically performed to answer a very narrow question in a restricted setting. To give an example, the work on automatically selecting music while doing sports, e.g. [43,45,6], is evaluated on the very question of whether pace or heartbeat of the user does synchronize with the tempo of the music. Likewise Kaminskas and Ricci's work on matching music with places of interest [32], even though it is evaluated by involving real users, comprises only the single question of whether the music suggested by their algorithm is suited for particular places of interest or not. Different dimensions of the relation between images and music are not addressed. Although this is perfectly fine for the intended use cases, such highly specific evaluation settings are not able to provide answers to more general questions of music retrieval and recommendation, foremost because these settings fail at offering *explanations* for the (un)suitability of the musical items under investigation.

An evaluation approach that tries to alleviate this shortcoming is presented in [7], where subjective listening tests to assess music recommendation algorithms are conducted using a multifaceted questionnaire. Besides investigating the enjoyment a user feels when listening to the recommended track ("liking"), the authors also ask for the user's "listening intention", whether or not the user knows the artist and song ("familiarity"), and whether he or she would like to request more similar music ("give-me-more"). A similar evaluation scheme is suggested by Firan et al. [22], though they only investigate liking and novelty.
In summary, almost all approaches reported are still more systems-based than user-centric.

2.2 What about the user in other communities?

Other research communities, in particular the Recommendation Systems (RS) and the Text-IR communities, include the user much more comprehensively in evaluation. An overview of relevant literature in these two areas is given below.

When looking at the RS community, there is a long tradition in using the systems-based performance measure of Root Mean Square Error (RMSE) to measure recommendation quality [50]. This measure is typically computed and investigated in leave-one-out experiments. However, a few years ago the RS community started to recognize the importance of user-centric evaluation strategies, and reacted accordingly. Pu and Chen in [49] present a highly detailed user-centric evaluation framework, which make use of psychometric user satisfaction questionnaires. They analyze a broad variety of factors organized into four categories: perceived system qualities, user beliefs, user attitudes, and behavioral intentions. In particular, Pu and Chen highlight (i) *perceived accuracy*, i.e. the degree to which users feel that the recommendations match their preferences, (ii) *familiarity*, i.e. whether users have previous knowledge about the recommended items, (iii) *novelty*, i.e. whether novel items are recommended, (iv) *attractiveness*, i.e. whether recommended items are capable of stimulating a positive emotion of interest or desire, (v) *enjoyability*, i.e. whether users have joyful experience with the suggested items, (vi) *diversity* of the recommended items, and (vii) *context compatibility*, i.e. whether the recommended items fit the current user context, such as the user's mood or activity. In addition to these aspects, Pu and Chen propose user questions that assess the *perceived usefulness* and *transparency* of a recommender, as well as *user intentions* towards the recommendation system.
A similar study, though not as comprehensive, is presented by Dooms et al. in [18]. The authors use a questionnaire and ask users to explicitly rate different qualities of the recommender system under investigation using a Likert-type 5-point scale. In addition, they also look into implicit user feedback, analyzing the user interaction with the system. Based on this input, Dooms et al. identify eight relevant aspects that are important to users when interacting with recommendation systems: match between recommended items and user

interests, familiarity of the recommended items, ability to discover new items, similarity between recommended items, explanation why particular items are recommended, overall satisfaction with the recommendations, trust in the recommender, and willingness to purchase some of the recommended items.
To further underline the importance RS researchers attribute to user-centric evaluation, a workshop series dedicated to the very topic of "User-centric Evaluation of Recommender Systems and Their Interface"[2], held in conjunction with the "ACM Conference on Recommender Systems", came into life in 2010 [35].

Some of the user-centric aspects addressed in RS literature can also be found in IR research. In particular, the properties of *novelty* and *diversity* [13] as well as *transparency* [55], i.e. explaining why a particular item has been returned by a retrieval systems, are frequently mentioned. Also the aspect of *redundancy*, i.e. omitting redundant results that are annoying for most users, is addressed [64].
The IR community is thus also seeing a paradigm shift in evaluation and performance measurement, away from the traditional systems-based relevance measures, such as precision, recall, precision at k retrieved documents (P@k), mean average precision (MAP), or discounted cumulative gain (DCG), e.g. [4], towards considering *user interaction* and system usage [3]. A vital role is hence played by emphasizing interaction with information, instead of passive user consumption of documents or items returned by a retrieval system [28]. Järvelin as well as Callan et al. [9] propose a shift in the general design of IR systems, away from the concept of users finding documents, towards information interaction via clustering, linking, summarizing, arranging, and social networks.

3 How to evaluate user-centric MIR?

In what follows we will argue that whereas evaluation of systems-based MIR has quite matured, evaluation of user-centric MIR is still in its infancy.

3.1 Systems-based and user-centric MIR experiments

Let us start by reviewing what the nature of experiments is in the context of MIR. The basic structure of MIR experiments is the same as in any other experimental situation: the objective is to measure the effect of different treatments on a dependent variable. Typical dependent variables in systems-based MIR are various performance measures like accuracy, precision, root mean squared error or training time; and the treatments are the different algorithms to evaluate and compare, or different parametrizations of the same algorithm. A standard computer experiment is genre classification, where the treatments are different types of classification algorithm, say algorithms A and B, and the dependent variable is the achieved accuracy. But there are many other factors that might influence the results of the algorithms. For example, the musical expertise of the end user plays an important role in how good genre classification algorithms are perceived: as mentioned, a Heavy Metal fan is able to distinguish between Viking Metal and Death Metal, while most people do not. As another example, consider a fan of Eric Clapton that wishes to find similar music and a recommender system suggests Cream or Derek and the Dominos, which are bands surely known by this specific user but rather not by every general user. Any factor that is able to influence the dependent variables should be part of the experimental design, such as the musical expertise or known artists in the examples above. The important thing to note is that for systems-based MIR, which uses only computer experiments, it is comparably easy to control all important factors which could have an influence on the dependent variables. This is because the number of factors is both manageable and controllable, since the experiments are being conducted on computers and not in the real world. Indeed, the only changing factor is the algorithm to use.

Already early in the history of MIR research, gaps concerning the evaluation of MIR systems have been identified. Futrelle and Downie [24], in their 2003 review of the first three years of the ISMIR conference, identify two major problems: (i) no commonly accepted means of comparing retrieval techniques, (ii) few, if any, attempts to study potential users of MIR systems. The first problem concerns the lack of standardized frameworks to evaluate computer experiments, while the second problem concerns the barely existing inclusion of users in MIR studies. Flexer [23], in his review of the 2004 ISMIR conference [8], argues for the necessity of statistical evaluation of MIR experiments. He presents minimum requirements concerning statistical evaluation by applying fundamental notions of statistical hypothesis testing to MIR research. But his discussion is concerned with systems-based MIR: the example used throughout the paper is that of automatic genre classification based on audio content analysis.

Statistical testing is needed to assess the confidence in that the observed effects on the dependent variables are caused by the varied independent variables and not by mere chance, i.e. to ascertain that the observed differences are too large to attribute them to random in-

[2] `http://ucersti.ieis.tue.nl`

fluences only. The MIR community is often criticized for the lack of statistical evaluation it performs, e.g., only two papers in the ISMIR 2004 proceedings [8] employ a statistical test to measure the statistical significance of their results. A first evaluation benchmark took place at the 2004 ISMIR conference [10] and ongoing discussions about evaluation of MIR experiments have led to the establishment of the annual evaluation campaign for MIR algorithms ("Music Information Retrieval Evaluation eXchange", MIREX) [20]. Starting with the MIREX 2006 evaluation [20], statistical tests have been used to analyze results in most tasks. But besides using the proper instruments to establish the statistical significance of results, it is equally important to make sure to control all important factors in the experimental design, always bearing in mind that statistical significance does not measure practical importance for users [56, 29].

In 2012, MIREX consisted of 15 tasks, such as audio classification, melody extraction, audio key detection to structural segmentation and audio tempo estimation. All these tasks follow a systems-based evaluation framework, in which we mainly measure different characteristics of the system response. The only user component included in these evaluations is the ground truth data, which usually consists of very low-level annotations such as beat marks, tempo, frequency, etc. The two exceptions that include a high-level form of ground truth, closer to a real-world setting, are *Audio Music Similarity and Retrieval* and *Symbolic Melodic Similarity*, in which human listeners provide annotations regarding the musical similarity between two music clips. But it is very important to realize that the real utility of a system for a real user goes far beyond these simple expected-output annotations and effectiveness measures, no matter how sophisticated they are [44, 27]. Systems-based evaluations, as of today, completely ignore user context and user properties, even though they clearly influence the result. For example, human assessors in the similarity tasks provide an annotation based on *their* personal and subjective notion of similarity. Do all users agree with that personal notion? Definitely not, and yet, we completely ignore this fact in our systems-based evaluations.

The situation concerning evaluation of user-centric MIR research is far less well developed. In a recent comprehensive review [58] of user studies in the MIR literature by Weigl and Guastavino, papers from the first decade of ISMIR conferences and related MIR publications were analyzed. A central result is that MIR research has a mostly systems-centric focus. Only twenty papers fell under the broad category of "user studies", which is an alarmingly small number given that 719 articles have been published in the ISMIR conference series alone. To make things worse, these user studies are "predominantly qualitative in nature" and of "largely exploratory nature" [58]. The explored topics range from user requirements and information needs, insights into social and demographic factors to user-generated meta-information and ground truth. This all points to the conclusion that evaluation of user-centric MIR is at its beginning and that especially a more rigorous quantitative approach is still missing.

3.2 A closer look at the music similarity tasks

In discussing the challenges of quantitative evaluation of user-centric MIR we like to turn to an illustrative example: the recent 2012 *Audio Music Similarity and Retrieval* (AMS) and *Symbolic Melodic Similarity* (SMS) tasks[3] within the annual MIREX [20] evaluation campaign. In the AMS task, each of the competing algorithms was given 50 random queries (5 from each of 10 different genres), while in the SMS task each system was given 30 queries. All systems had to rank the songs in a collection (7000 30-second-audio-clips in AMS and 5274 melodies in SMS) according to their similarity to each of the query songs. The top 10 songs ranked for each query were then evaluated by human graders. For each individual (query, candidate)-pair, a single human grader provided both a Fine score (from 0 to 100) and a Broad score (not similar, somewhat similar, or very similar) indicating how similar the songs were in *their* opinion. The objective here is again to compare all systems (the treatments); the dependent variable is the aggregated score of the subjects' Broad and Fine appraisal of the perceived similarity. From these scores over a sample of queries, we estimate the expected effectiveness of each system for an arbitrary query, and determine which systems are better accordingly.

But since this is a real-world experiment involving human subjects, there is a whole range of additional factors that influence the results. For instance, there are social and demographic factors that might clearly influence the user's judgment of music similarity: their age, gender, cultural background, and especially their musical history, experience, and knowledge. But also factors concerning their momentary situation during the actual listening experiment might have an influence: time of day, mood, physical condition, etc. Not to forget more straightforward variables like type of speakers or headphones used for the test. It is clear that all these variables influence the perceived similarity between two

[3] The MIREX 2012 results and details can be found at `http://www.music-ir.org/mirex/wiki/2012`.

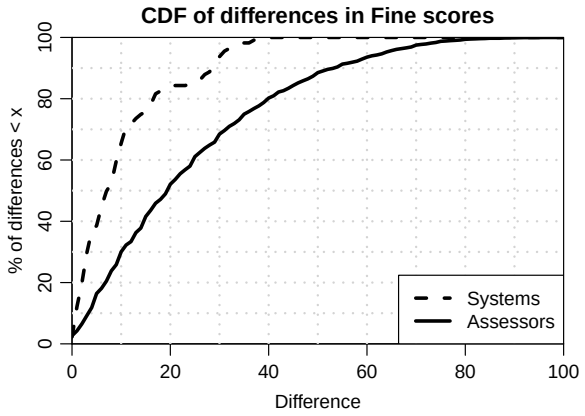


Fig. 2 Distribution of differences among MIREX 2006 AMS assessors and among participating systems in 2006, 2007, 2009, 2010, 2011 and 2012. Differences among assessors are larger than differences among systems.

songs and thus the system comparisons, but none of them is considered in the experiments.

In the 2006 run of MIREX, three different assessors provided similarity annotations for the AMS and SMS tasks [30]. As expected, there were wide differences between assessors, most probably due to their different context and background characteristics. As Figure 2 shows, over 50% of the times there was a difference larger than 20 between the Fine scores given by two of the AMS assessors, and even large differences over 50 were observed more than 10% of the times. This indicates that differences between end users can be quite large, which is particularly worrying considering that observed differences among systems are much smaller (e.g., the difference between the best and worst 2012 systems was just 17, again according to the Fine scale). In fact, recent work established that as much as 20% of the users are not satisfied by system outputs which were supposed to be "perfect" according to the systems-based evaluations [56]. That is, as much as 20% improvement could be achieved if we included the user context and user properties as part of our queries so that systems personalize their outputs. But what we actually do in these experiments is ignore these user effects, so we should at best consider our human assessors as a sample from a wider population[4]. As such, we can only interpret our results as the expected performance of the systems, not only for an arbitrary query, but also for an arbitrary user. If we want to evaluate our systems in a more realistic setting, we must change the queries from "what songs are similar to this one" to "what songs are similar to this one, *if we target a user like this*".

As mentioned in Section 1, even the choice of dependent variable is debatable. After all, what does "similar" really mean in the context of music? Timbre, mood, harmony, melody, tempo, etc. might all be valid criteria for different people to assess similarity. This points to a certain lack of rigor concerning the instruction of subjects during the experiment. Also, is similarity the only variable we should measure? The system–user interaction can be characterized with many more variables, some of which are not related to similarity at all (e.g., system response time, ease of use or interface design) [26]. Furthermore, the relationship between a system-measure and a user-measure might not be as we expect. For instance, it has been shown that relatively small differences in systems-based measures such as similarity are not even noticed by end users, questioning the immediate practical significance of small improvements and showing the need for systems-based measures that more closely capture the user response [56].

This enumeration of potential problems is not intended to badmouth these MIREX tasks, which still are a valuable contribution and an applaudable exception to the rule of low-level, nearly algorithm-only evaluation. But it is meant as a warning, to highlight the explosion of variables and factors that might add to the variance of observed results and might obscure significant differences. In principle, all such factors have to be recorded at the least, and provided to the systems for better user-aware evaluations. If MIR is to succeed in maturing from purely systems-based to user-centric research, we will have to leave the nice and clean world of our computers and face the often bewilderingly complex real world of real human users and all the challenges this entails for proper design and evaluation of experiments. To make this happen it will be necessary that our community, with a predominantly engineering background, opens up to the so-called "soft sciences" of psychology and sociology, for instance, which have developed instruments and methods to deal with the complexity of human subjects.

4 What should we do?

Incorporating real users in both the development and assessment of music retrieval systems is of course an expensive and arduous task. However, recent trends in music distribution, in particular the emergence of music streaming services that make available millions of tracks to their users, call for intelligent, personalized and context-aware systems to deal with this abundance. Concerning the development of such

[4] Even though this is likely not the case in the MIREX AMS and SMS tasks as the judgments are certainly biased towards that of music researchers and scientists.

systems, we believe that the following three reasons have prevented major breakthroughs so far: (i) a general lack of research on user-centric systems, (ii) a lack of awareness of the limitations and usefulness of systems-based evaluation, (iii) the complexity and cost of evaluating user-centric systems. In designing such systems, the user should already be taken into account at an early stage during the development process, and play a larger role in the evaluation process as well. We need to better understand what the user's individual requirements are and address these requirements in our implementations. Otherwise, it is unlikely that even the spiffiest personalized systems will succeed (without frustrating the user). We hence identify the following four key requirements for elaborating user-centric music retrieval systems:

User models that encompass different social scopes are needed. They may aggregate an individual model, an interest group model, a cultural model, and a global model. Furthermore, the user should be modeled as comprehensively as possible, in a fine-grained and multifaceted manner. With today's sensor-packed smartphones, other intelligent devices, and frequent use of social media it has become easy to perform extensive context logging. Of course, privacy issues must also be taken seriously.
Learning more about the real user needs, such as information or entertainment need is vital to elaborate respective user models. To give some examples of aspects that may contribute to these needs, Pu and Chen [49] and Schedl et al. [53] mention, among others, similarity and diversity, familiarity, novelty, trendiness, attractiveness, serendipity, popularity, enjoyability, and context compatibility.

Personalization aspects have to be taken into account. In this context, it is important to note the highly subjective, cognitive component in the understanding of music and judgment of its personal appeal. Therefore, designing user-aware music applications requires intelligent machine learning and information retrieval techniques, in particular, preference learning approaches that relate the user context to concise and situation-dependent music preferences.

Multifaceted similarity measures that combine different feature categories (music content, music context, user context, and user properties) are required. The corresponding representation models should then not only allow to derive similarity between music via content-related aspects, such as beat strength or instruments playing, or via music context-related properties, such as the geographic origin of the performer or a song's lyrics, but also to describe users and user groups in order to compute listener-based features and similarity scores. Based on these user-centric information, novel personalized and context-aware music recommender systems, retrieval algorithms, and music browsing interfaces will emerge.

Evaluation of user-centric music retrieval approaches should include in the experimental design all independent variables that are able to influence the dependent variables. In particular, such factors may well relate to individual properties of the human assessors, which may present problems of both practical and computational nature.
Furthermore, it is advisable to make use of recent approaches to minimize the amount of labor required by the human assessors, while at the same time maintaining the reliability of the experiments. This can be achieved, for instance, by employing the concept of "Minimal Test Collections" (MTC) [11] in the evaluation of music retrieval systems [57].

The idea of MTC is that there is no need to let users judge all items retrieved for a particular query in order to estimate with high confidence which of two systems performs better. Analyzing which queries (and retrieval results) are the most discriminative in terms of revealing performance differences between two systems, it is shown in [57] that the number of user judgments can be reduced considerably for evaluating music retrieval tasks.
When looking at user-centric evaluation in fields related to MIR, it seems that in particular the Text-IR and Recommendation Systems communities, are already a step further. They especially foster the use of evaluation strategies that result in highly specific qualitative feedback on user satisfaction and similar subjective demands, for instance in [49,18]. Such factors are unfortunately all too frequently forgotten in MIR research. We should hence broaden our view by looking into how other communities address the user, investigate which strategies can also be applied to our tasks, and what we can thus borrow from these communities. For example, user aspects reported in [49,18,53] include perceived similarity, diversity, familiarity, novelty, trendiness, attractiveness, serendipity, popularity, enjoyability, transparency, and usefulness. We presume that at least some of these also play an important role in music retrieval and should thus be considered in user-centered evaluation of MIR systems.

By paying attention to these advices, we are sure that the exciting field of user-centric music infor-

mation retrieval will continue to grow and eventually provide us with algorithms and systems that offer personalized and context-aware access to music in an unintrusive way.

References

1. Aucouturier, J.J., Bigand, E.: Mel Cepstrum & Ann Ova: The Difficult Dialog Between MIR and Music Cognition. In: Proc. ISMIR (2012)
2. Aucouturier, J.J., Pachet, F.: Representing Musical Genre: A State of the Art. Journal of New Music Research **32**(1), 83–93 (2003)
3. Azzopardi, L., Järvelin, K., Kamps, J., Smucker, M.D.: Report on the SIGIR 2010 Workshop on the Simulation of Interaction. SIGIR Forum **44**(2), 35–47 (2011)
4. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval – the concepts and technology behind search, 2nd edn. Addison-Wesley, Pearson, Harlow, England (2011)
5. Baltrunas, L., Kaminskas, M., Ludwig, B., Moling, O., Ricci, F., Lüke, K.H., Schwaiger, R.: InCarMusic: Context-Aware Music Recommendations in a Car. In: Proc. EC-Web (2011)
6. Biehl, J.T., Adamczyk, P.D., Bailey, B.P.: DJogger: A Mobile Dynamic Music Device. In: CHI 2006: Extended Abstracts (2006)
7. Bogdanov, D., Herrera, P.: How Much Metadata Do We Need in Music Recommendation? A Subjective Evaluation Using Preference Sets. In: Proc. ISMIR (2011)
8. Buyoli, C., Loureiro, R.: Fifth International Conference on Music Information Retrieval. Universitat Pompeu Fabra (2004). URL `http://books.google.at/books?id=r0BXAAAACAAJ`
9. Callan, J., Allan, J., Clarke, C.L.A., Dumais, S., Evans, D.A., Sanderson, M., Zhai, C.: Meeting of the MINDS: An Information Retrieval Research Agenda. SIGIR Forum **41**(2), 25–34 (2007)
10. Cano, P., Gómez, E., Gouyon, F., Herrera, P., Koppenberger, M., Ong, B., Serra, X., Streich, S., Wack, N.: ISMIR 2004 Audio Description Contest (2006)
11. Carterette, B., Allan, J., Sitaraman, R.: Minimal Test Collections for Retrieval Evaluation. In: Proc. SIGIR, pp. 268–275. Seattle, WA, USA (2006)
12. Cebrián, T., Planagumà, M., Villegas, P., Amatriain, X.: Music Recommendations with Temporal Context Awareness. In: Proc. RecSys (2010)
13. Clarke, C., Kolla, M., Cormack, G., Vechtomova, O., Ashkan, A., Büttcher, S., MacKinnon, I.: Novelty and Diversity in Information Retrieval Evaluation. In: Proc. SIGIR, pp. 659–666. Singapore (2008)
14. Cunningham, S., Caulder, S., Grout, V.: Saturday Night or Fever? Context-Aware Music Playlists. In: Proc. Audio Mostly (2008)
15. Cunningham, S.J., Bainbridge, D., Falconer, A.: More of an Art than a Science: Supporting the Creation of Playlists and Mixes. In: Proc. ISMIR, pp. 474–477. Victoria, Canada (2006)
16. Cunningham, S.J., Downie, J.S., Bainbridge, D.: "The Pain, The Pain": Modelling Music Information Behavior And The Songs We Hate. In: Proc. ISMIR, pp. 474–477. London, UK (2005)
17. Cunningham, S.J., Jones, M., Jones, S.: Organizing Digital Music for Use: An Examination of Personal Music Collections. In: Proc. ISMIR, pp. 447–454. Barcelona, Spain (2004)
18. Dooms, S., De Pessemier, T., Martens, L.: A User-centric Evaluation of Recommender Algorithms for an Event Recommendation System. In: Proc. RecSys: Workshop on Human Decision Making in Recommender Systems (Decisions@RecSys'11) and User-Centric Evaluation of Recommender Systems and Their Interfaces (UCERSTI 2), pp. 67–73. Chicago, IL, USA (2011)
19. Dornbush, S., English, J., Oates, T., Segall, Z., Joshi, A.: XPod: A Human Activity Aware Learning Mobile Music Player. In: Proc. Workshop on Ambient Intelligence, IJCAI (2007)
20. Downie, J.S.: The Music Information Retrieval Evaluation eXchange (MIREX). D-Lib Magazine (2006). URL `http://dlib.org/dlib/december06/downie/12downie.html`
21. Elliott, G.T., Tomlinson, B.: PersonalSoundtrack: Context-aware Playlists that Adapt to User Pace. In: CHI 2006: Extended Abstracts (2006)
22. Firan, C.S., Nejdl, W., Paiu, R.: The Benefit of Using Tag-Based Profiles. In: Proceedings of the 5th Latin American Web Congress (LA-WEB), pp. 32–41. Santiago de Chile, Chile (2007)
23. Flexer, A.: Statistical Evaluation of Music Information Retrieval Experiments. Journal of New Music Research **35**(2), 113–120 (2006)
24. Futrelle, J., Downie, J.S.: Interdisciplinary Research Issues in Music Information Retrieval: ISMIR 2000-2002. Journal of New Music Research **32**(2), 121–131 (2003)
25. Hargreaves, D.J., MacDonald, R., Miell, D.: Musical Communication, chap. How do people communicate using music? Oxford University Press (2005)
26. Hu, X., Kando, N.: User-centered Measures vs. System Effectiveness in Finding Similar Songs. In: Proc. ISMIR. Porto, Portugal (2012)
27. Ingwersen, P., Järvelin, K.: The Turn: Integration of Information Seeking and Retrieval in Context. Springer (2005)
28. Järvelin, K.: IR Research: Systems, Interaction, Evaluation and Theories. SIGIR Forum **45**(2), 17–31 (2012)
29. Johnson, D.: The Insignificance of Statistical Significance Testing. The Journal of Wildlife Management pp. 763–772 (1999)
30. Jones, M.C., Downie, J.S., Ehmann, A.F.: Human Similarity Judgments: Implications for the Design of Formal Evaluations. In: Proc. ISMIR. Vienna, Austria (2007)
31. Kamalzadeh, M., Baur, D., Möller, T.: A Survey on Music Listening and Management Bahaviours. In: Proc. ISMIR, pp. 373–378. Porto, Portugal (2012)
32. Kaminskas, M., Ricci, F.: Location-Adapted Music Recommendation Using Tags. In: Proc. UMAP (2011)
33. Knees, P., Pohle, T., Schedl, M., Widmer, G.: A Music Search Engine Built upon Audio-based and Web-based Similarity Measures. In: Proc. SIGIR (2007)
34. Knees, P., Widmer, G.: Searching for Music Using Natural Language Queries and Relevance Feedback. In: Proc. AMR (2007)
35. Knijnenburg, B.P., Schmidt-Thieme, L., Bollen, D.G.: Workshop on User-centric Evaluation of Recommender Systems and Their Interfaces. In: Proc. RecSys, pp. 383–384. Barcelona, Spain (2010)
36. Kohonen, T.: Self-Organizing Maps, *Springer Series in Information Sciences*, vol. 30, 3rd edn. Springer, Berlin, Germany (2001)
37. Koren, Y., Bell, R., Volinsky, C.: Matrix Factorization Techniques for Recommender Systems. Computer **42** (2009)

Markus Schedl, Gerhard Widmer, Peter Knees, Tim Pohle

A Music Information System Automatically Generated via Web Content Mining Techniques

Information Processing & Management, 47, 2011

Information Processing and Management 47 (2011) 426–439

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

A music information system automatically generated via Web content mining techniques

Markus Schedl *, Gerhard Widmer, Peter Knees, Tim Pohle

Department of Computational Perception, Johannes Kepler University, Altenberger Straße 69, A-4040 Linz, Austria

ARTICLE INFO

Article history:
Received 2 April 2009
Received in revised form 23 August 2010
Accepted 6 September 2010
Available online 8 October 2010

Keywords:
Music information retrieval
Web content mining
Information systems
Application
Evaluation

ABSTRACT

This article deals with the problem of *mining music-related information from the Web* and representing this information via a *music information system*. Novel techniques have been developed as well as existing ones refined in order to automatically gather information about music artists and bands. After searching, retrieval, and indexing of Web pages that are related to a music artist or band, *Web content mining* and *music information retrieval* techniques were applied to capture 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*. Approaches to extracting these pieces of information are presented and evaluation experiments are described that investigate the proposed approaches' performance. From the insights gained by the various experiments an *Automatically Generated Music Information System* (AGMIS) providing Web-based access to the extracted information has been developed. AGMIS demonstrates the feasibility of automated music information systems on a large collection of more than 600,000 music artists.

1. Introduction and context

Over the past few years, digital music distribution via the World Wide Web has seen a tremendous increase. As a result, music-related information beyond the pure digital music file (musical meta-data) is becoming more and more important as users of online music stores nowadays expect to be offered such additional information. Moreover, digital music distributors are in need of such additional value that represents a decisive advantage over their competitors.

Also music information systems, i.e., systems primarily focusing on *providing information* about music, not on selling music, typically offer multimodal information about music artists,[1] albums, and tracks (e.g., genre and style, similar artists, biographies, song samples, or images of album covers). In common music information systems, such information is usually collected and revised by experts, e.g., *All Music Guide* (amg, 2009) or relies on user participation, e.g., *last.fm* (las, 2009). In contrast, this paper describes methods for building such a system by automatically extracting the required information from the Web at large. To this end, various techniques to estimate relations between artists, to determine descriptive terms, to extract band members and instrumentation, and to find images of album covers were elaborated, evaluated, refined, and aggregated.

Automatically retrieving information about music artists is an important task in music information retrieval (MIR), cf. Downie (2003). It permits, for example, enriching music players with meta-information (Schedl, Pohle, Knees, & Widmer, 2006c), automatically tagging of artists (Eck, Bertin-Mahieux, & Lamere, 2007), automatic biography generation (Alani et al., 2003), developing user interfaces to browse music collections by more sophisticated means than the textual browsing

* Corresponding author. Tel.: +43 (0) 732 2468 1512; fax: +43 (0) 732 2468 1520.
E-mail address: markus.schedl@jku.at (M. Schedl).

[1] In the following, we use the term "artist" to refer to both single musicians and bands.

0306-4573/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ipm.2010.09.002

facilities, in an artist – album – track hierarchy, traditionally offered (Knees, Schedl, Pohle, & Widmer, 2006; Pampalk & Goto, 2007), or defining similarity measures between artists. Music similarity measures can then be used, for example, to create relationship networks (Cano & Koppenberger, 2004), for automatic playlist generation (Aucouturier & Pachet, 2002; Pohle, Knees, Schedl, Pampalk, & Widmer, 2007), or to build music recommender systems (Celma & Lamere, 2007; Zadel & Fujinaga, 2004) or music search engines (Knees, Pohle, Schedl, & Widmer, 2007).

In the following, an overview of existing Web mining techniques for MIR is given in Section 2. Section 3 briefly presents the methods developed and refined by the authors, together with evaluation results. Section 4 describes the application of the techniques from Section 3 for creating the *Automatically Generated Music Information System* (AGMIS), a system providing information on more than 600,000 music artists. Finally, in Section 5, conclusions are drawn, and directions for future work are pointed out.

2. Related work

Related work mainly consists of methods to derive similarities between music artists and attribute descriptive terms to an artist, which is also known as *tagging*. Traditionally, similarities between songs or artists are calculated on some kind of musically relevant features extracted form the audio signal. Such features usually aim at capturing *rhythmic* or *timbral* aspects of music. Rhythm is typically described by some sort of *beat histogram*, e.g., Pampalk, Rauber, and Merkl (2002) and Dixon, Gouyon, and Widmer (2004 et al.), whereas timbral aspects are usually approximated by *Mel Frequency Cepstral Coefficients* (MFCCs), e.g., Aucouturier, Pachet, and Sandler (2005) and Mandel and Ellis (2005). However, such audio signal-based similarity measures cannot take into account aspects like the cultural context of an artist, the semantics of the lyrics of a song, or the emotional impact of a song on its listener. In fact, the performance of such purely audio-based measures seems to be limited by a "glass ceiling", cf. Aucouturier and Pachet (2004).

Overcoming this limitation requires alternative methods, most of which have in common the *participation of lots of people* to form a large information resource. Like typical Web 2.0 applications, such methods benefit from the wisdom of the crowd. The respective data is hence often called *cultural features* or *community meta-data*. Probably the most prominent example of such features are those gained in a collaborative tagging process. Lamere (2008) gives a comprehensive overview of the power of social tags in the music domain, shows possible applications, but also outlines shortcomings of collaborative tagging systems. Celma (2008) laboriously analyzed and compared different tagging approaches for music, especially focusing on their use for music recommendation and taking into account the long tail of largely unknown artists.

Cultural features were, however, already used in MIR before the Web 2.0-era and the emergence of folksonomies. Early approaches inferring music similarity from sources other than the audio signal use, e.g., co-occurrences of artists or tracks in radio station playlists and compilation CDs (Pachet, Westerman, & Laigre, 2001) or in arbitrary lists extracted from Web pages (Cohen & Fan, 2000). Other researchers extracted different term sets from artist-related Web pages and built individual term profiles for each artist (Ellis, Whitman, Berenzweig, & Lawrence, 2002; Knees, Pampalk, & Widmer, 2004; Whitman & Lawrence, 2002). The principal shortcoming of such similarities inferred from cultural features is their restriction to the artist level since there is usually too little data available on the level of individual songs. The most promising approach to transcend these limitations is combining multiple features extracted from different sources. For example, a method that enriches Web-based with audio-based features to create term profiles at the track level is proposed in Knees, Pohle, et al. (2007). The authors present a search engine to retrieve music by textual queries, like "rock music with great riffs". Pohle et al. (2007) present an approach to automatic playlist generation that approximates the solution to a Traveling Salesman Problem on signal-based distances, but uses Web-based similarities to direct the search heuristics.

As for determining descriptive terms for an artist, such as instruments, genres, styles, moods, emotions, or geographic locations, Pampalk, Flexer, and Widmer (2005) use a self-assembled dictionary and apply different term weighting techniques on artist-related Web pages to assign terms to sets of artists and cluster them in a hierarchical manner. The term weighting functions analyzed were based on document frequency (DF), term frequency (TF), and term frequency · inverse document frequency (TF·IDF) variations. The conducted experiments showed that considering only the terms in the dictionary outperforms using the unpruned, complete set of terms extracted from the Web pages. Geleijnse and Korst (2006) and Schedl et al. (2006c) independently present an approach to artist tagging that estimates 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 mention the artist name. The calculated probabilities are used to predict the most probable value of attributes related to artist or music (e.g., *happy*, *neutral*, *sad* for the attribute *mood*). Both papers particularly try to categorize artists according to their genre, which seems reasonable as genre names are also among the most frequently applied tags in common music information systems like *last.fm* (Geleijnse, Schedl, & Knees, 2007). Another category of tagging approaches make use of *last.fm* tags and distill certain kinds of information. For example, Hu, Bay, and Downie (2007) use a part-of-speech (POS) tagger to search *last.fm* tags for adjectives that describe the mood of a song. Eck et al. (2007) use the machine learning algorithm AdaBoost to learn relations between acoustic features and *last.fm* tags.

A recent approach to gathering tags is the so-called *ESP games* (von Ahn & Dabbish, 2004). These games provide some form of incentive[2] to the human player to solve problems that are hard to solve for computers, e.g., capturing emotions evoked

[2] Commonly the pure joy of gaming is enough to attract players.

when listening to a song. Turnbull, Liu, Barrington, and Lanckriet (2007), Mandel and Ellis (2007), and Law, von Ahn, Dannenberg, and Crawford (2007) present such game-style approaches that provide a fun way to gather musical annotations.

3. Mining the Web for music artist-related information

All methods proposed here rely on the availability of artist-related data on the Web. The authors' principal approach to extracting such data is the following. Given only a list of artist names, we first query a search engine[3] to retrieve the URLs of up to 100 top-ranked search results for each artist. The content available at these URLs is extracted and stored for further processing. To overcome the problem of artist names that equal common speech words and to direct the search towards the desired information, we use task-specific query schemes like `"band name" + music + members` to obtain data related to band members and instrumentation. We do not account for multilingual pages by varying the language of the additional keywords (e.g., `"music"`, `"Musik"`, `"musique"`, `"musica"`) as this would considerably increase the number of queries issued to the search engine. It has to be kept in mind, however, that restricting the search space to English pages might yield undiscovered pages which are nevertheless relevant to the artist. In any case, this approach relies on the ranking algorithm of the search engine.

Depending on the task to solve, either a *document-level inverted index* or a *word-level index* (Zobel & Moffat, 2006) is then created from the retrieved Web pages. In some cases, especially when it comes to artist tagging, a special dictionary of musically relevant terms is used for indexing. After having indexed the Web pages, we gain artist-related information of various kinds as described in the following.

As an alternative approach to the use of a search engine for Web page selection, we could use a focused crawler (Chakrabarti, van den Berg, & Dom, 1999) trained to retrieve pages from the music domain. We are currently assessing this alternative as it would avoid relying on commercial search engines and would allow us to build a corpus specific to the music domain. On the other hand, companies like *Google* offer a huge corpus which can be accessed very efficiently. Thus, we still have to compare these two strategies (directed search using a search engine vs. focused crawling) and assess their performance in depth, which will be part of future work.

3.1. Relations between artists

3.1.1. Similarity Relations

A key concept in music information retrieval and crucial part of any music information system is *similarity relations* between artists. To model such relations, we propose an approach that is based on co-occurrence analysis (Schedl, Knees, & Widmer, 2005a). More precisely, the similarity between two artists i and j is inferred from the conditional probability that the artist name i occurs on a Web page that was returned as response to the search query for the artist name j and vice versa. The formal definition of the similarity measure is given in Formula (1), where I represents the set of Web pages returned for artist i and $df_{i,J}$ is the document frequency of the artist name i calculated on the set of Web pages returned for artist j.

$$sim_{cooc}(i,j) = \frac{1}{2} \cdot \left(\frac{df_{i,J}}{|J|} + \frac{df_{j,I}}{|I|} \right) \tag{1}$$

Having calculated the similarity for each pair of artists in the input list, it is possible to output, for any artist, a list of most similar artists, i.e., building a recommender system. Evaluation in an artist-to-genre classification task using a *k-nearest neighbor classifier* on a set of 224 artists from 14 genres yielded accuracy values of about 85% averaged over all genres, cf. Schedl et al. (2005a).

3.1.2. Prototypicality relations

Co-occurrences of artist names on Web pages (together with genre information) can also be used to derive information about the *prototypicality of an artist for a certain genre* (Schedl, Knees, & Widmer, 2005b, 2006). To this end, the asymmetry of the one-sided, co-occurrence-based similarity measure is exploited as explained below. Taking a look at Formula (1) again and focusing on the single terms $\frac{df_{i,J}}{|J|}$ and $\frac{df_{j,I}}{|I|}$ that estimate the single probability for an artist name to be found on the page retrieved for another artist, it is obvious that, in general, $\frac{df_{i,J}}{|J|} \neq \frac{df_{j,I}}{|I|}$. Such asymmetric similarity measures have some disadvantages, the most important of which is that they do not allow to induce a metric in the feature space. Moreover, they produce unintuitive and hard to understand visualizations when using them to build visual browsing applications based on clustering, like the *nepTune* interface (Knees, Schedl, Pohle, & Widmer, 2007). However, the asymmetry can also be beneficially exploited for deriving artist popularity or prototypicality of an artist for a certain genre (or any other categorical aspect). Taking into account the asymmetry of the co-occurrence-based similarity measure, the main idea behind our approach is that it is more likely to find the name of a well-known and representative artist for a genre on many Web pages about a lesser known artist, e.g., a newcomer band, than vice versa. To formalize this idea, we developed an approach that is based on the *backlink/forward link-ratio* of two artists i and j from the same genre, where a *backlink* of i from j is defined as any occurrence of artist i on a Web page that is known to contain artist j, whereas a *forward link* of i to j is defined as any

[3] We commonly used *Google* (goo, 2009), but also experimented with *exalead* (exa, 2009).

occurrence of j on a Web page known to mention i. Relating the number of forward links to the number of backlinks for each pair of artists from the same genre, a ranking of the artist prototypicality for the genre under consideration is obtained. More precisely, we count the number of forward links and backlinks on the document frequency-level, i.e., all occurrences of artist name i on a particular page retrieved for j contribute 1 to the backlink count of i, regardless of the term i's frequency on this page. To alleviate the problem of artist names being highly ranked due to their resemblance to common speech words,[4] we use a correction factor that penalizes artists whose prototypicality is exorbitantly, therefore unjustifiably, high for all genres. Putting this together, the refined prototypicality ranking function $r(i,g)$ of artist i for genre g is given in Formula (2), where G represents the set of artists in genre g. The penalization term is given in Formula (3), where A denotes the set of all artists in the collection. The functions $bl(i,j)$ and $fl(i,j)$ as defined in Formulas (4) and (5), respectively, measure whether the number of backlinks of i from j, as defined above, exceeds the number of forward links of i to j (in this case, $bl(i,j) = 1$ and $fl(i,j) = 0$) or the number of backlinks of i from j is equal or less than the number of forward links of i from j (in this case, $bl(i,j) = 0$ and $fl(i,j) = 1$). $df_{j,I}$ gives the number of Web pages retrieved for artist i that also mention artist j. This number hence represents a document frequency and equals the respective term in Formula (1). $|I|$ is the total number of pages retrieved for artist i. The normalization function $\|\cdot\|$ shifts all values to the positive range and maps them to $[0,1]$.

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

$$penalty(i) = \left\| \log\left(\frac{\sum_{j\in A}^{j\neq i} fl(i,j) + 1}{\sum_{j\in A}^{j\neq i} bl(i,j) + 1} \right) \right\|^2 \tag{3}$$

$$bl(i,j) = \begin{cases} 1 & \text{if } \frac{df_{j,I}}{|I|} < \frac{df_{i,J}}{|J|} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

$$fl(i,j) = \begin{cases} 1 & \text{if } \frac{df_{j,I}}{|I|} \geqslant \frac{df_{i,J}}{|J|} \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

We conducted an evaluation experiment using a set of 1995 artists from 9 genres extracted from *All Music Guide*. As ground truth we used the so-called "tiers" that reflect the importance, quality, and relevance of an artist to the respective genre, judged by *All Music Guide*'s editors, cf. amgabout (2007). Calculating *Spearman's rank-order correlation*, e.g., Sheskin (2004), between the ranking given by Formula (2) and the ranking given by *All Music Guide*'s tiers, revealed an average correlation coefficient of 0.38 over all genres. More details on the evaluation can be found in Schedl, Knees, and Widmer (2006).

To give an example of how the penalization term influences the ranking, we first consider the band "Tool", which is classified as "Heavy Metal" by *All Music Guide*'s editors.[5] This band has a backlink/forward link-ratio of $\frac{263}{8} = 32.875$ when applying Formula (2) without the $penalty(i)$ term. As a result, "Tool" ranks 3rd in the prototypicality ranking for the genre "Heavy Metal" (only superseded by "Death" and "Europe"), which we and also *All Music Guide*'s editors believe does not properly reflect the band's true importance for the genre, even though "Tool" is certainly no unknown band to the metal aficionado. However, when multiplying the ratio with the penalization term, which is 0.1578 for "Tool" (according to Formula (3)), the band is downranked to rank number 29 (of 271), which seems more accurate. In contrast, the artist "Alice Cooper", who obviously does not equal a common speech word, has a backlink/forward link-ratio of $\frac{247}{24} = 10.29$, which translates to rank 10. With a value of 0.8883 for Formula (3), "Alice Cooper" still remains at the 10th rank after applying the penalization factor, which we would judge highly accurate.

3.2. Band member and instrumentation detection

Another type of information indispensible for a music information system is *band members and instrumentation*. In order to capture such aspects, we first apply to the Web pages retrieved for a band a named entity detection (NED) approach. To this end, we extract all 2-, 3-, and 4-grams, assuming that the complete name of any band member does comprise of at least two and at most four single names. We then discard all n-grams whose tokens contain only one character and retain only the n-grams with their first letter in upper case and all other letters in lower case. Finally, we use the *iSpell English Word Lists* (isp, 2006) to filter out all n-grams where at least one token equals a common speech word. This last step in the NED is essential to suppress noise in the data, since in Web pages, word capitalization is used not only to denote named entities, but often also for highlighting purposes. The remaining n-grams are regarded as potential band members.

Subsequently, we perform shallow linguistic analysis to obtain the actual instrument(s) of each member. To this end, a set of seven patterns, like "`M`, the `R`" or "`M` plays the `I`", where M is the potential member, I is the instrument, and R is the member's role in the band, is applied to the n-grams and the surrounding text as necessary. For I and R, we use lists of synonyms to cope with the use of different terms for the same concept (e.g., "drummer" and "percussionist"). We then calculate the

[4] Terms like *Kiss*, *Bush*, or *Hole* often occur on (artist-related) Web pages, but do not necessarily denote the respective bands.

[5] In this example, we use the same data set of 1995 artists as in Schedl, Knees, and Widmer (2006).

document frequencies of the patterns and accumulate them over all seven patterns for each (M,I)-tuple. In order to suppress uncertain information, we filter out those (M,I)-pairs whose document frequency falls below a dynamic threshold t_f, which is parametrized by a constant f. t_f is expressed as a fraction f of the highest document frequency of any (M,I)-pair for the band under consideration. Consider, for example, a band whose top-ranked singer, according to the DF measure, has an accumulated DF count of 20. Using $f = 0.06$, all potential members with an aggregated DF of less than 2 would be filtered out in this case as $t_{0.06} = 20 \cdot 0.06 = 1.2$. The remaining tuples are predicted as members of the band under consideration. Note that this approach allows for an m:n assignment between instruments and bands.

An evaluation of this approach was conducted on a data set of 51 bands with 499 members (current and former ones). The ground truth was gathered from *Wikipedia* (wik, 2009), *All Music Guide*, *discogs* (dis, 2009), or the band's Web site. We also assessed different query schemes to obtain *Google*'s top-ranked Web pages for each band:

- "`band`" + `music` (abbr. *M*)
- "`band`" + `music` + `review` (abbr. *MR*)
- "`band`" + `music` + `members` (abbr. *MM*)
- "`band`" + `music` + `lineup` (abbr. *LUM*)

Varying the parameter f, we can adjust the trade-off between precision and recall, which is depicted in Fig. 1. From the figure, we can see that the query schemes *M* and *MM* outperform the other two schemes. Another finding is that f values in the range $[0.2, 0.25]$ (depending on query scheme) maximize the sum of precision and recall, at least for the used data set. Considering that there exists an upper limit for the recall achievable with our approach, due to the fact that usually not all band members are covered by the fetched 100 Web pages per artist, these results are pretty promising. The upper limit for the recall for the various query schemes is: *M*: 53%, *MR*: 47%, *MM*: 56%, *LUM*: 55%. For more details on the evaluation, a comprehensive discussion of the results, and a second evaluation taking only current band members into account, the interested reader is invited to consider Schedl and Widmer (2007).

3.3. Automatic tagging of artists

We perform automatically attributing textual descriptors to artists, commonly referred to as *tagging*, using a dictionary of about 1500 musically relevant terms in the indexing process. This dictionary resembles the one used in Pampalk et al. (2005). It contains terms somehow related to music, e.g., names of musical instruments, genres, styles, moods, time periods, and geographical locations. The dictionary is available at http://www.cp.jku.at/people/schedl/music/cob_terms.txt.

As for term selection, i.e., finding the most descriptive terms for an artist, we investigated three different term weighting measures (DF, TF, and TF·IDF) in a quantitative user study using a collection of 112 well-known artists (14 genres, 8 artists each), cf. Schedl and Pohle (2010). To this end, the 10 most important terms according to each term weighting function had been determined. In order to avoid biasing of the results, the 10 terms obtained by each weighting function were then merged into one list per artist. Hence, every participant was presented a list of 112 artist names and, for each name, the corresponding term list. Since the authors 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 associate every term to one of the three classes + (*good description*), − (*bad description*), and ∼ (indifferent or not wrong, but not a description specific for the artist). We had five participants in the user study and received a total of 172 assessments. Mapping the ratings in class + to the value 1, those in class − to −1, and those in class ∼ to 0 and calculating the arithmetic mean of the values of all

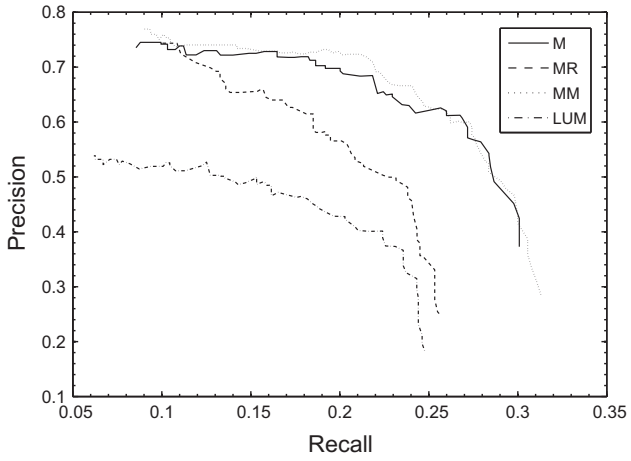


Fig. 1. Precision/recall-plot of the approach to band member and instrumentation detection.

Table 1
Results of Friedman's test to assess the significance of the differences in the term weighting measures.

N	92
df	2
χ^2	16.640
p	0.00000236

assessments for each artist, we obtained a score representing the average excess of the number of good terms over the number of bad terms. These scores were 2.22, 2.43, and 1.53 for TF, DF, and TF·IDF, respectively.

To test for the significance of the results, we performed *Friedman's two-way analysis of variance* (Friedman & March, 1940; Sheskin, 2004). This test is similar to the two-way ANOVA, but does not assume a normal distribution of the data. It is hence a non-parametric test, and it requires related samples (ensured by the fact that for each artist all three measures were rated). The outcome of the test is summarized in Table 1. Due to the very low p value, we can state that the variance differences in the results are significant with a very high probability. To assess which term weighting measures produce significantly different results, we conducted pairwise comparison between the results given by the three weighting functions. To this end, we employed the *Wilcoxon signed ranks test* (Wilcoxon, 1945) and tested for a significance level of 0.01. The test 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. This result is quite surprising as TF·IDF is a well-established term weighting measure and commonly used to describe text documents according to the vector space model, cf. Salton, Wong, and Yang (1975). A possible explanation for the worse performance of TF·IDF is that this measure assigns high weights to terms that are very specific for a certain artist (high TF and low DF), which is obviously a desired property when it comes to distinguish one artist from another. In our application scenario, however, we aim at finding the most descriptive terms – not the most discriminative ones – for a given artist. This kind of terms seems to be better determined by the simple TF and DF measures. Hence, for the AGMIS application, we opted for the DF weighting to automatically select the most appropriate tags for each artist.

3.4. Co-Occurrence Browser

To easily access the top-ranked Web pages of any artist, we designed a user interface called *Co-Occurrence Browser* (COB), cf. Fig. 2. COB is based on the *Sunburst* visualization technique (Andrews & Heidegger, 1998; Stasko & Zhang, 2000), which we brought to the third dimension. The purpose of COB is threefold: First, it facilitates getting an overview of the set of Web pages related to an artist by structuring and visualizing them according to co-occurring terms. Second, it reveals meta-information about an artist through the descriptive terms extracted from the artist's Web pages. Third, by extracting the multimedia contents from the set of the artist's Web pages and displaying them via the COB, the user can explore the Web pages by means of audio, image, and video data.

In short, based on the dictionary used for automatic tagging, COB groups the Web pages of the artist under consideration with respect to co-occurring terms and ranks the resulting groups by their document frequencies.[6] The sets of Web pages are then visualized using the approach presented in Schedl, Knees, Widmer, Seyerlehner, and Pohle (2007). In this way, COB allows for browsing the artist's Web pages by means of descriptive terms. Information on the amount of multimedia content is encoded in the arcs' height, where each Sunburst visualization accounts for a specific kind of multimedia data. Thus, in Fig. 2, the top-most Sunburst represents the video content, the middle one the image content, and the lower one the audio content found on the respective Web pages.

3.5. Album cover retrieval

We presented preliminary attempts to automatically retrieve album cover artwork in Schedl, Knees, Pohle, and Widmer (2006). For the article at hand, we refined our approach and conducted experiments with content-based methods (using image processing techniques) as well as with context-based methods (using text mining) for detecting images of album covers on the retrieved Web pages. The best performing strategy, which we therefore employed to build AGMIS, uses the text distance between artist and album name and ⟨img⟩ tag as indicator for the respective image's likelihood of showing the sought album cover. To this end, we create a *word-level index* (Zobel & Moffat, 2006) that does not only contain the plain text, but also the HTML tags of the retrieved Web pages. After having filtered all images that are unlikely to show an album cover, as described below, we output the image with minimum distance between ⟨img⟩ tag and artist name and ⟨img⟩ tag and album name on the set of Web pages retrieved for the artist under consideration. Formally, the selection function is given in Formula (6), where $pos_i(t)$ denotes the offset of term t, i.e., its position i in the Web page p, and P_a denotes all pages retrieved for artist a.

[6] Any term weighting measure can be used, but the simple DF measure seemed to capture the most relevant terms best, cf. Section 3.3.

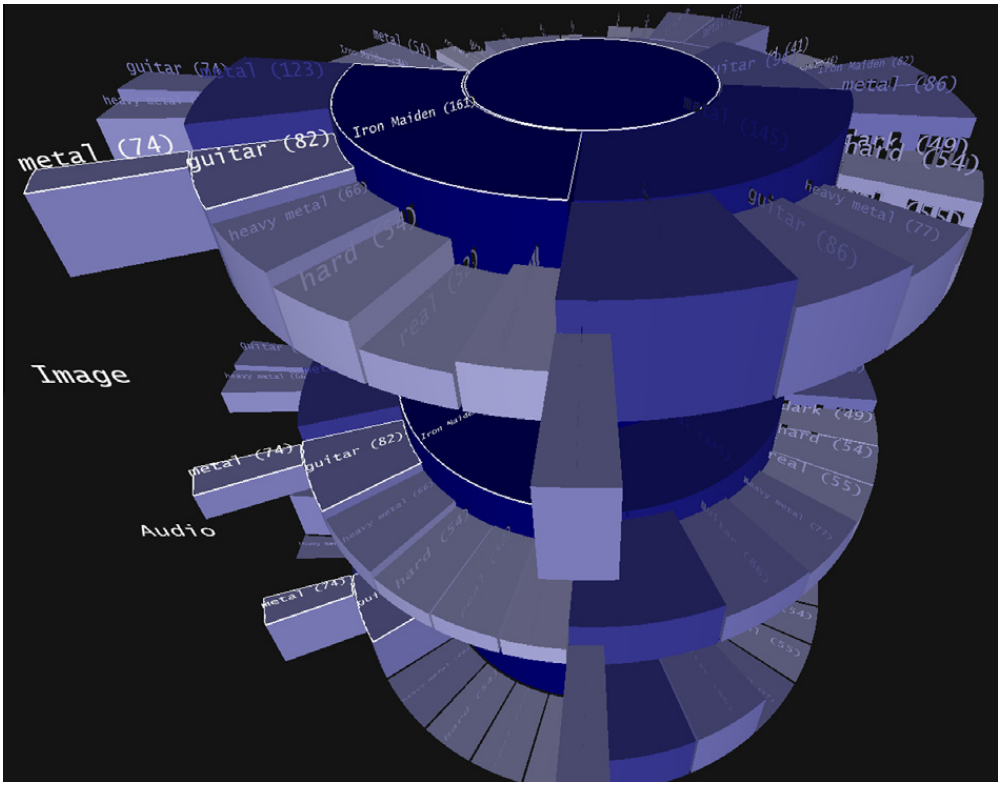


Fig. 2. COB visualizing a collection of Web pages retrieved for the band *Iron Maiden*.

$$min_{i,j,k}|pos_i(\langle img\rangle tag) - pos_j(artist\ \ name)| + |pos_i(\langle img\rangle tag) - pos_k(album\ \ name)| \quad \forall p \in P_a \tag{6}$$

As for filtering obviously erroneous images, content-based analysis is performed. Taking the almost quadratic shape of most album covers into account, all cover images that have non-quadratic dimensions within a tolerance of 15% are rejected. Since images of scanned compact discs often score highly on the text distance function, we use a circle detection technique to filter out those false positives. Usually, images of scanned discs are cropped to the circle-shaped border of the compact disc, which allows to use a simple circle detection algorithm. 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 RGB color histograms. 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.

On a test collection of 255 albums by 118 distinct, mostly European and American artists, our approach achieved a precision of up to 89% at a recall level of 93%, precision being defined as the number of correctly identified cover images among all predicted images, recall being defined as the number of found images among all albums in the collection. On a more challenging collection of 3311 albums by 1593 artists from all over the world, the approach yielded precision values of up to 73% at a recall level of 80%.

4. An automatically generated music information system

Since we aimed at building a music information system with broad artist coverage, we first had to gather a sufficiently large list of artists, on which the methods described in the previous section were applied. To this end, we extracted from *All Music Guide* nearly 700,000 music artists, organized in 18 different genres. In a subsequent data preprocessing step, all artists that were mapped to identical strings after *non-character removal*[7] were discarded, except for one occurrence. Table 2 lists the genre distribution of the remaining 636,475 artists according to *All Music Guide*, measured as absolute number of artists in each genre and as percentage in the complete collection. The notably high number 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 *All Music Guide* seemed the most reasonable solution to obtain a real-world artist list.

[7] This filtering was performed to cope with ambiguous spellings for the same artist, e.g., "B.B. King" and "BB King".

Table 2
List of genres used in AGMIS with the corresponding number of artists and their share in the complete collection as well as the number of artists for which no Web pages were found (*0-PC*).

Genre	Artists	%	*0-PC*	%
Avantgarde	4469	0.70	583	13.05
Blues	13,592	2.14	2003	14.74
Celtic	3861	0.61	464	12.02
Classical	11,285	1.77	1895	16.79
Country	16,307	2.56	2082	12.77
Easy listening	4987	0.78	865	17.35
Electronica	35,250	5.54	3101	8.80
Folk	13,757	2.16	2071	15.05
Gospel	26,436	4.15	5597	21.17
Jazz	63,621	10.00	10,866	17.08
Latin	33,797	5.31	9512	28.14
New age	13,347	2.10	2390	17.91
Rap	26,339	4.14	2773	10.53
Reggae	8552	1.34	1320	15.43
RnB	21,570	3.39	2817	13.06
Rock	267,845	42.08	39,431	14.72
Vocal	11,689	1.84	1988	17.01
World	59,771	9.39	17,513	29.30
Total	636,475	100.00	107,271	16.85

The sole input to the following data acquisition steps is the list of extracted artist names, except for the prototypicality estimation (cf. Section 3.1.2), which also requires genre information, and for the determination of album cover artwork (cf. Section 3.5), which requires album names. This additional information was also extracted from *All Music Guide*.

An overview of the data processing involved in building AGMIS is given in Fig. 3. The data acquisition process can be broadly divided into the three steps *querying* the search engine for the URLs of artist-related Web pages, *fetching* the HTML documents available at the retrieved URLs, and *indexing* the content of these documents.

Querying. We queried the *exalead* search engine for URLs of up to 100 top-ranked Web pages for each artist in the collection using the query scheme `"artist name" NEAR music`. The querying process took approximately one month. Its outcome was a list of 26,044,024 URLs that had to be fetched next.

Fetching. To fetch this large number of Web pages, we 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. It yielded a total of 732.6 gigabytes of Web pages.

Some statistics concerning the retrieved Web pages give interesting insights. Table 2 shows, for each genre, the number of artists for which not a single Web page could be determined by the search engine, i.e., artists with a page count of zero. Not very surprisingly, the percentage is highest for the genres "Latin" and "World" (nearly 30% of zero-page-count-artists), which comprise many artists known only in regions of the world that are lacking broad availability of Internet access. In contrast, a lot of information seems to be available for artists in the genres "Electronica" and "Rap" (about 10% of 0-PC-artists). Table 3 depicts the number of Web pages retrieved for all artists per genre (column *RP*), the arithmetic mean of Web pages retrieved for an artist (column RP_{mean}), and the number of retrieved pages with a length of zero, i.e., pages that were empty or could not be fetched for some reason. Since the main reason for the occurrence of such pages were server errors, their relative frequencies are largely genre-independent, as it can be seen in the fifth column of Table 3. The table further shows the median and arithmetic mean of the page counts returned by *exalead* for the artists in each genre. These values give strong indication that artists in the genres "Latin", "Gospel", and "World" tend to be underrepresented on the Web.

Indexing. To create a *word-level index* of the retrieved Web pages (Zobel & Moffat, 2006), the open source indexer *Lucene Java* (luc, 2008) was taken as a basis and adapted by the authors to suit the HTML format of the input documents and the requirements for efficiently extracting the desired artist-related pieces of information.

Although indexing seems to be a straightforward task at first glance, we had to resolve certain issues. Foremost some heavily erroneous HTML files were encountered, which caused *Lucene* to hang or crash, and thus required special handling. More precisely, some HTML pages showed a size of tens of megabytes, but were largely filled with escape characters. 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.

AGMIS makes use of two indexes. Creating the first one was performed applying neither stopping, nor stemming, nor casefolding as it is used for band member and instrumentation detection (cf. Section 3.2) and to calculate artist similarities (cf. Section 3.1.1). Since the patterns applied in the linguistic analysis step of our approach to band member detection contain a lot of stop words, applying stopping either would have been virtually useless (when using a stop word list whose entries were corrected for the words appearing in the patterns) or would have yielded a loss of information crucial to the application of the patterns. Since artist names sought for in our approach to similarity estimation typically also contain stop words, applying stopping would be counterproductive for this purpose as well. The size of the optimized, compressed first index is 228 gigabytes. A second index containing only the terms in the music dictionary was created to generate term

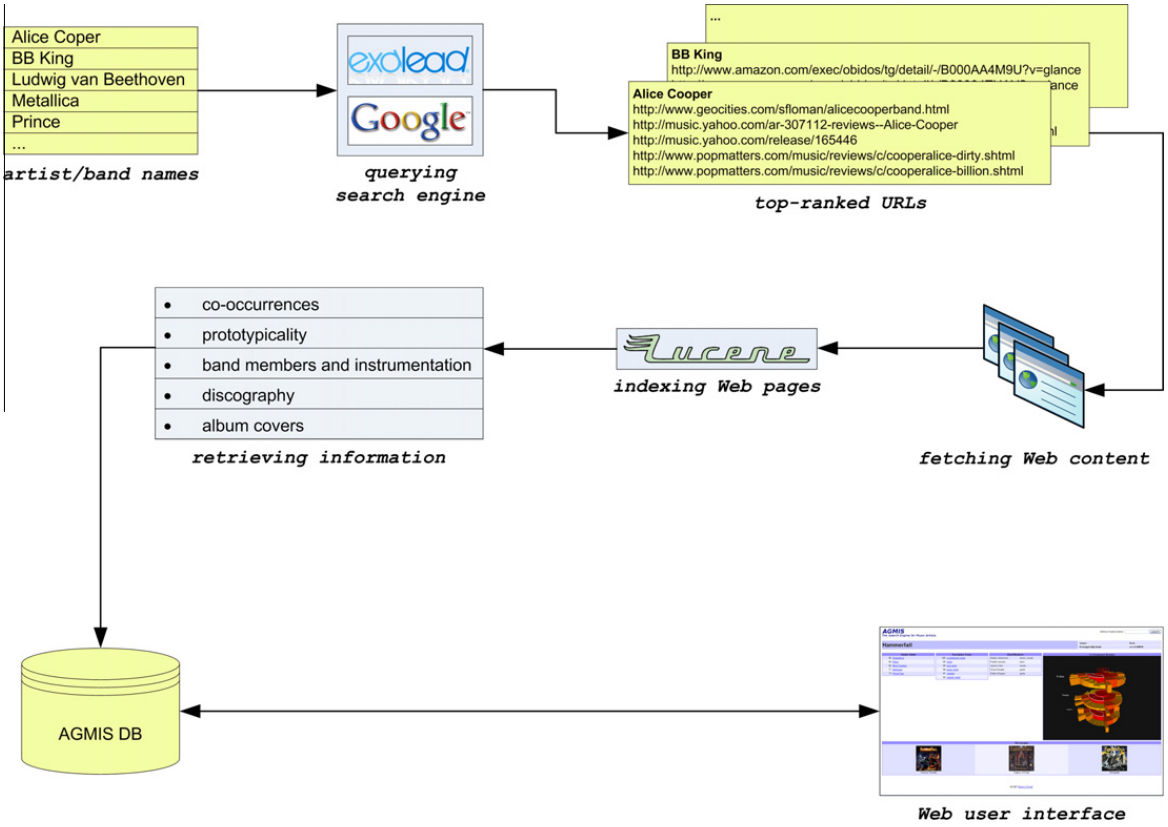


Fig. 3. Data processing diagram of AGMIS.

Table 3
The number of retrieved Web pages per genre (*RP*) and its mean per artist (RP_{mean}), the number of empty Web pages among them (*0-L*), and the median and mean of available Web pages according to the page-count-value returned by the search engine (PC_{med} and PC_{mean}).

Genre	RP	RP_{mean}	*0-L*	%	PC_{med}	PC_{mean}
Avantgarde	204,870	46	32,704	15.96	29	14,969
Blues	554,084	40	89,832	16.21	18	2893
Celtic	136,244	35	23,627	17.34	25	5415
Classical	509,269	45	99,181	19.48	27	4149
Country	696,791	42	116,299	16.69	22	2562
Easy listening	187,749	37	32,758	17.45	14	4808
Electronica	1,973,601	56	317,863	16.11	65	31,366
Folk	544,687	39	89,385	16.41	18	5166
Gospel	876,017	33	142,690	16.29	8	4791
Jazz	2,306,785	36	361,160	15.66	13	6720
Latin	866,492	25	139,660	16.12	4	19,384
New age	488,799	36	82,075	16.79	13	12,343
Rap	1,322,187	50	223,052	16.87	37	38,002
Reggae	377,355	44	58,180	15.42	22	16,000
RnB	898,787	41	141,339	15.73	17	17,361
Rock	12,058,028	43	1,908,904	15.83	21	16,085
Vocal	461,374	39	77,073	16.71	15	10,421
World	1,577,769	26	257,649	16.33	4	14,753
Total	26,040,888	40	4,193,431	16.10	16	15,120

profiles for the purpose of artist tagging (cf. Section 3.3) and for the COB (cf. Section 3.4). The size of this index amounts to 28 gigabytes.

4.1. AGMIS' user interface

The pieces of information extracted from the artist-related Web pages and inserted into a relational *MySQL* (mys, 2008) database are offered to the user of AGMIS via a Web service built 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

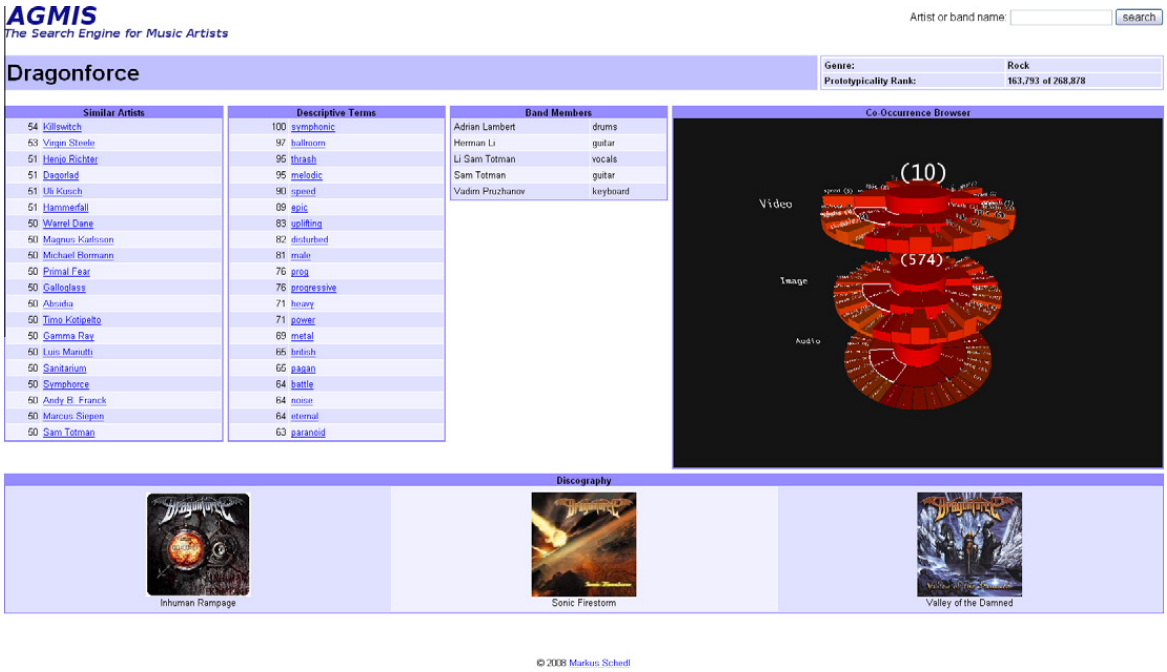


Fig. 4. The user interface provided by AGMIS for the band *Dragonforce*.

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, Ravikumar, & Fienberg, 2003). The user is then provided a list of approximately matching artist names, from which he or she can select one.

After the user has selected the desired artist, AGMIS delivers an artist information page. Fig. 4 shows an example of such a page for the band *Dragonforce*. 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 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. To the right of the lists described so far, the Co-Occurrence Browser is integrated into the user interface as a *Java Applet* to permit browsing the indexed Web pages and their multimedia content. 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.

4.2. Computational complexity

Most tasks necessary to build AGMIS were quite time-consuming. The querying, fetching, and indexing processes, the creation of artist term profiles, the calculation of term weights, and all information extraction tasks were performed on two standard personal computers with *Pentium 4* processors clocked at 3 GHz, 2 GB RAM, and a RAID-5 storage array providing 2 TB of usable space. In addition, a considerable amount of external hard disks serving as temporary storage facilities were required.

4.2.1. Running times

In Table 4, precise running times for indexing, information extraction, and database operation tasks are shown for those tasks for which we measured the time. Calculating the artist similarity matrix was carried out as follows. Computing the complete 636,475 $\times$ 636,475 similarity matrix requires 202,549,894,575 pairwise similarity calculations. 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 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, we first split the 636,475 $\times$ 636,475 matrix into 50 rows and 50 columns, yielding 1275 submatrices when storing symmetric elements only once. Each 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.

4.2.2. Asymptotic runtime complexity

The asymptotic runtime complexities of the methods presented in Section 3 are summarized in Table 5, supposing that querying, fetching, and indexing was already performed. Querying is obviously linear (in terms of issued requests) in the

Table 4
Some running times of tasks performed while creating AGMIS.

Task	Time (s)
Creating *Lucene* index using all terms (no stopping, stemming, casefolding)	218,681
Creating *Lucene* index using the music dictionary	211,354
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 (audio, image, video) 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 database	356,195
Importing the 20 top-ranked terms for each artist into the AGMIS database	3649
Importing album names and covers into the AGMIS database	6686

Table 5
Asymptotic runtime complexities of the IE approaches.

Task	Runtime complexity
Artist similarity calculation	$\mathcal{O}(n^2 \cdot \log k)$
Artist prototypicality estimation	$\mathcal{O}(n^2 \cdot \log k)$
Band member and instumentation detection	$\mathcal{O}(n \cdot k \cdot p)$
Artist tagging	$\mathcal{O}(n \cdot k)$
Album cover retrieval	$\mathcal{O}(n \cdot k)$

number of artists, i.e., $\mathcal{O}(n)$, provided that the desired number of top-ranked search results p retrieved per artist does not exceed the number of results that can be returned by the search engine in one page. Fetching can be performed in $\mathcal{O}(n \cdot p)$, but will usually require less operations (cf. Table 2, the average number of Web pages retrieved per artist is 40). Using a *B-tree* (Bayer, 1971) as data structure, indexing can be performed in $\mathcal{O}(t \cdot \log t)$, where t is the total number of terms to be processed.

In Table 5, n denotes the total number of artists and k the total number of keys in the index. Creating the symmetric similarity matrix and estimating the prototypicality for each artist both require n^2 requests to the index. Since each request takes $\log k$, the complexity of the whole process is $\mathcal{O}(n^2 \cdot \log k)$. The band member detection requires k operations to extract the potential band members, i.e., n-grams, for each of which p operations are needed to evaluate the patterns and obtain their document frequencies, p being the number of patterns in all variations, i.e., all synonyms for instruments and roles counted as a separate pattern (cf. Section 3.2). In total, the asymptotic runtime complexity is therefore $\mathcal{O}(n \cdot k \cdot p)$. The automatic artist tagging procedure is in $\mathcal{O}(n \cdot k)$, where k is again the number of terms in the index. However, as we use a dedicated index for the purpose of artist tagging, $k \approx 1500$, and therefore $k \ll n$. Finally, the current implementation of our album cover retrieval technique requires $n \cdot k$ operations, since all keys in the index have to be sought for ⟨img⟩ tags, artist names, and album names. This could be sped up by building an optimized index with clustered ⟨img⟩ tags, which will be part of future work.

5. Conclusions and future work

This article has given an overview of state-of-the-art techniques for Web-based information extraction in the music domain. In particular, techniques to mine relations between artists (similarities and prototypicality), band members and instrumentation, descriptive terms, and album covers were presented. Furthermore, this article briefly described the *Co-Occurrence Browser* (COB), a user interface to organize and access artist-related Web pages via important, music-related terms and multimedia content. It was further shown that the proposed approaches can be successfully applied on a large scale using a real-world database of more than 600,000 music artists. Integrating the extracted information into a single information system yielded the *Automatically Generated Music Information System* (AGMIS), whose purpose is to provide access to the large amount of data gathered. The design, implementation, and feeding of the system were reported in detail.

Even though the evaluation experiments conducted to assess the techniques underlying AGMIS showed promising results, they still leave room for improvement in various directions. First, Web page retrieval could be pursued using focused crawling instead of directed search via search engines. This would presumably yield more accurate results, while at the same time limit Web traffic. Second, 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, temporal information would allow for creating band and artist histories as well as time-dependent relationship networks. Automatically generated biographies would be the ultimate aim. Finally, the information gathered by the Web mining techniques presented here could be

complemented with information extracted from the audio signal. Audio signal-based similarity information at 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, cf. Korst and Geleijnse (2006) and Knees et al. (2005). Employing methods to align audio and lyrics could eventually even allow for applications like an automatic karaoke system.

Acknowledgments

This research is supported by the *Austrian Science Fund (FWF)* under Project Numbers L511-N15, Z159, and P22856-N23. The authors would further like to thank *Julien Carcenac* from *exalead* for his support in the querying process.

References

Alani, H., Kim, S., Millard, D. E., Weal, M. J., Hall, W., Lewis, P. H., et al (2003). Automatic ontology-based knowledge extraction from web documents. *IEEE Intelligent Systems, 18*(1), 14–21.

amgabout (2007). <http://www.allmusic.com/cg/amg.dll?p=amg&sql=32:amg/info_pages/a_about.html> Accessed November 2007.

amg (2009). <http://www.allmusic.com> Accessed November 2009.

Andrews, K., & Heidegger, H. (1998). Information slices: Visualising and exploring large hierarchies using cascading, semi-circular discs. In *Proceedings of IEEE information visualization 1998*. Research Triangle Park, NC, USA.

Aucouturier, J.-J., & Pachet, F. (2002). Scaling up music playlist generation. In *Proceedings of the IEEE international conference on multimedia and expo (ICME 2002)* (pp. 105–108). Lausanne, Switzerland.

Aucouturier, J.-J., & Pachet, F. (2004). Improving timbre similarity: How high is the sky? *Journal of Negative Results in Speech and Audio Sciences, 1*(1).

Aucouturier, J.-J., Pachet, F., & Sandler, M. (2005). The way it sounds: Timbre models for analysis and retrieval of music signals. *IEEE Transactions on Multimedia, 7*(6), 1028–1035.

Bayer, R. (1971). Binary B-trees for virtual memory. In *Proceedings of the ACM SIG FIDET workshop*. San Diego, CA, USA.

Cano, P., & 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)* (pp. 466–469). Barcelona, Spain.

Celma, O. (2008). Music recommendation and discovery in the long tail. Ph.D. Thesis, Universitat Pompeu Fabra, Barcelona, Spain. <http://mtg.upf.edu/ocelma/PhD/doc/ocelma-thesis.pdf>.

Celma, O., & Lamere, P. (2007). ISMIR 2007 tutorial: Music recommendation. <http://mtg.upf.edu/~ocelma/MusicRecommendationTutorial-ISMIR2007> Accessed December 2007.

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

Cohen, W. W., Ravikumar, P., & 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)* (pp. 73–78). Acapulco, Mexico.

Cohen, W. W., & Fan, W. (2000). Web-collaborative filtering: Recommending music by crawling the web. *WWW9/Computer Networks, 33*(1–6), 685–698.

dis (2009). <http://www.discogs.com> Accessed October 2009.

Dixon, S., Gouyon, F., & Widmer, G. (2004). Towards characterisation of music via rhythmic patterns. In *Proceedings of the 5th international symposium on music information retrieval (ISMIR 2004)* (pp. 509–516). Barcelona, Spain.

Downie, J. S. (2003). Music information retrieval. *Annual Review of Information Science and Technology, 37*, 295–340.

Eck, D., Bertin-Mahieux, T., & Lamere, P. (2007). Autotagging music using supervised machine learning. In *Proceedings of the 8th international conference on music information retrieval (ISMIR 2007)*. Vienna, Austria.

Ellis, D. P., Whitman, B., Berenzweig, A., & 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.

exa (2009). <http://www.exalead.com> Accessed February 2009.

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.

Geleijnse, G., & Korst, J. (2006). Web-based artist categorization. In *Proceedings of the 7th international conference on music information retrieval (ISMIR 2006)*. Victoria, Canada.

Geleijnse, G., Schedl, M., & 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.

goo (2009). <http://www.google.com> Accessed March 2009.

Hu, X., Bay, M., & 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.

isp (2006). <http://wordlist.sourceforge.net> Accessed June 2006.

Knees, P., Pampalk, E., & Widmer, G. (2004). Artist classification with web-based data. In *Proceedings of the 5th international symposium on music information retrieval (ISMIR 2004)* (pp. 517–524). Barcelona, Spain.

Knees, P., Schedl, M., & Widmer, G. (2005). Multiple lyrics alignment: automatic retrieval of song lyrics. In *Proceedings of 6th international conference on music information retrieval (ISMIR 2005)* (pp. 564–569). London, UK.

Knees, P., Schedl, M., Pohle, T., & Widmer, G. (2006). 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, P., Pohle, T., Schedl, M., & Widmer, G. (2007). 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, P., Schedl, M., Pohle, T., & Widmer, G. (2007). Exploring music collections in virtual landscapes. *IEEE MultiMedia, 14*(3), 46–54.

Korst, J., & Geleijnse, G. (2006). Efficient lyrics retrieval and alignment. In W. Verhaegh, E. Aarts, W. ten Kate, J. Korst, & S. Pauws (Eds.), *Proceedings of the 3rd Philips symposium on intelligent algorithms (SOIA 2006)* (pp. 205–218). Eindhoven, the Netherlands.

Lamere, P. (2008). Social tagging and music information retrieval. *Journal of New Music Research: Special Issue: From Genres to Tags: Music Information Retrieval in the Age of Social Tagging, 37*(2), 101–114.

las (2009). <http://last.fm> Accessed February 2009.

Law, E., von Ahn, L., Dannenberg, R., & Crawford, M. (2007). Tagatune: A game for music and sound annotation. In *Proceedings of the 8th international conference on music information retrieval (ISMIR 2007)*. Vienna, Austria.

luc (2008). <http://lucene.apache.org> Accessed January 2008.

Mandel, M. I., & 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.

Mandel, M. I., & Ellis, D. P. (2007). A web-based game for collecting music metadata. In *Proceedings of the 8th international conference on music information retrieval (ISMIR 2007)*. Vienna, Austria.
mys (2008). <http://www.mysql.com> Accessed June 2008.
Pachet, F., Westerman, G., & 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.
Pampalk, E., & Goto, M. (2007). MusicSun: A new approach to artist recommendation. In *Proceedings of the 8th international conference on music information retrieval (ISMIR 2007)*. Vienna, Austria.
Pampalk, E., Rauber, A., & Merkl, D. (2002). Content-based organization and visualization of music archives. In *Proceedings of the 10th ACM international conference on multimedia (MM 2002)* (pp. 570–579). Juan les Pins, France.
Pampalk, E., Flexer, A., & 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.
Pohle, T., Knees, P., Schedl, M., Pampalk, E., & Widmer, G. (2007). Reinventing the wheel: A novel approach to music player interfaces. *IEEE Transactions on Multimedia, 9*, 567–575.
Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM, 18*(11), 613–620.
Schedl, M., & Pohle, T. (2010). Enlightening the sun: A user interface to explore music artists via multimedia content. *Multimedia Tools and Applications: Special Issue on Semantic and Digital Media Technologies, 49*(1), 101–118.
Schedl, M., & 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, M., Knees, P., & 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, M., Knees, P., & 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, M., Knees, P., Pohle, T., & Widmer, G. (2006). Towards automatic retrieval of album covers. In *Proceedings of the 28th European conference on information retrieval (ECIR 2006)*. London, UK.
Schedl, M., Knees, P., & Widmer, G. (2006). 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, M., Pohle, T., Knees, P., & 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, M., Knees, P., Widmer, G., Seyerlehner, K., & Pohle, T. (2007). 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.
Sheskin, D. J. (2004). *Handbook of parametric and nonparametric statistical procedures* (3rd ed.). Boca Raton, London, New York, Washington, DC: Chapman & Hall/CRC.
Stasko, J., & Zhang, E. (2000). Focus + context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *Proceedings of IEEE information visualization 2000*. Salt Lake City, UT, USA.
Turnbull, D., Liu, R., Barrington, L., & Lanckriet, G. (2007). A game-based approach for collecting semantic annotations of music. In *Proceedings of the 8th international conference on music information retrieval (ISMIR 2007)*. Vienna, Austria.
von Ahn, L., & Dabbish, L. (2004). Labeling images with a computer game. In *CHI'04: Proceedings of the SIGCHI conference on human factors in computing systems*. New York, NY, USA: ACM Press.
Whitman, B., & Lawrence, S. (2002). Inferring descriptions and similarity for music from community metadata. In *Proceedings of the 2002 international computer music conference (ICMC 2002)* (pp. 591–598). Göteborg, Sweden.
wik (2009). <http://www.wikipedia.org> Accessed December 2009.
Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin, 1*(6), 80–83.
Zadel, M., & 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, J., & Moffat, A. (2006). Inverted files for text search engines. *ACM Computing Surveys, 38*, 1–56.

Markus Schedl holds a Ph.D. in computer science (computational perception) from the *Johannes Kepler University Linz*, where he is employed as assistant professor. He graduated in computer science from the *Vienna University of Technology*. His main research interests include Web mining, multimedia information retrieval, information visualization, and intelligent user interfaces.

Gerhard Widmer is professor and head of the *Department of Computational Perception* at the *Johannes Kepler University Linz* and head of the *Intelligent Music Processing and Machine Learning* group at the *Austrian Research Institute for Artificial Intelligence*, Vienna, Austria. He holds M.Sc. degrees from the *Vienna University of Technology* and the *University of Wisconsin*, Madison, and a Ph.D. in computer science from the *Vienna University of Technology*. His research interests are in machine learning, pattern recognition, and intelligent music processing. In 2009, he was awarded Austria's highest research prize, the Wittgenstein Prize, for his work on AI and music.

Peter Knees graduated in computer science. Since February 2005 he has been working as a project assistant at the *Johannes Kepler University Linz*. He performs research towards a doctoral thesis with a focus on music information retrieval. Since 2004 he has been studying psychology at the *University of Vienna*.

Tim Pohle holds a Dipl.-Inf. degree (equivalent to M.Sc. in computer science) from the *Technical University Kaiserslautern* and a Ph.D. in computer science from the *Johannes Kepler University Linz*. His main field of interest is music information retrieval with a special emphasis on audio based-techniques.

Markus Schedl, Tim Pohle, Noam Koenigstein, Peter Knees

What's Hot? Estimating Country-Specific Artist Popularity

Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR)

Utrecht, the Netherlands, August 2010

WHAT'S HOT ? ESTIMATING COUNTRY-SPECIFIC ARTIST POPULARITY

Markus Schedl[1], **Tim Pohle**[1], **Noam Koenigstein**[2], **Peter Knees**[1]

[1] Department of Computational Perception
Johannes Kepler University, Linz, Austria

[2] Faculty of Engineering
Tel Aviv University, Tel Aviv, Israel

ABSTRACT

Predicting artists that are popular in certain regions of the world is a well desired task, especially for the music industry. Also the cosmopolitan and cultural-aware music aficionado is likely be interested in which music is currently "hot" in other parts of the world. We therefore propose four approaches to determine *artist popularity rankings* on the country-level. To this end, we mine the following data sources: *page counts from Web search engines*, *user posts on Twitter*, *shared folders on the Gnutella file sharing network*, and *playcount data from last.fm*. We propose methods to derive artist rankings based on these four sources and perform cross-comparison of the resulting rankings via overlap scores. We further elaborate on the advantages and disadvantages of all approaches as they yield interestingly diverse results.

1. INTRODUCTION

To determine popular artists for a certain country or cultural region of the world, one can obviously look into publicly available music charts, such as the "Billboard Hot 100", released weekly for the United States of America by the *Billboard Magazine* [6]. However, this straightforward strategy is hardly feasibly when we aim at broaden the scope to the whole world. The reasons are manifold.

First, not all countries do release music charts for various reasons. Causes may be, for example, a lack of capability to determine music sales or an underdevelopment of music distribution at large. Even if data is available, it is often not publicly accessible, and even if so, not always in an easy-to-use format, e.g., via a Web service.
Second, even if charts are available for a specific country, they often cover only certain ways of music distribution. Commonly they are strongly biased towards sales figures of music albums. In some countries, however, they also include digital music sales via online stores. This inhomogeneity between countries, i.e., the in- or exclusion of certain distribution channels, make such data hardly comparable between different countries of the world. Another aspect to be considered here are possible heavy distortions caused by (illegal) music sharing channels, since legislation in this area varies severely between countries. In fact, the majority of today's music distribution is affected via file sharing networks [2]. Thus, traditional charts, such as the "Billboard Hot 100", are becoming less and less relevant.
Third, if the aim is to come up with a list of the most popular artists ever, countries lacking solid historical records constitute an obvious problem.

Summarizing these challenges, we conclude that analyzing which kind of music is popular in a specific country or cultural region necessitates taking a deeper look into various distribution channels and data sources. In this paper, we therefore present four different approaches to estimate artist popularity rankings on the country-level, each of which makes use of a different data source. The first one is based on *page count estimates* of Web search engines, the second approach analyzes *Twitter posts*, the third one derives information from meta-data of *users' shared folders in a Peer-to-Peer network*, and the fourth one uses *playcount data from last.fm*.

In the remainder of this paper we review related literature (Section 2), present four approaches to determine artist popularity on the country-level (Section 3), elaborate on the conducted evaluation experiments and discuss their results (Section 4), and finally draw conclusions (Section 5).

2. RELATED WORK

Related work falls into two categories: literature that particularly tackles the task of chart prediction, and work that relates to the four approaches we propose for this task.

Targeting the problem of predicting music charts, Koenigstein and Shavitt [26] present an approach to predict the charts based on search queries issued within the Peer-to-Peer (P2P) file sharing network *Gnutella* [35]. The authors show that a song's popularity in the P2P network highly correlates with its ranking in the Billboard charts. The authors' approach can further predict upcoming charts with high accuracy. However, for their analysis Koenigstein and Shavitt only consider the United States.
Pachet and Roy [33] try to predict the popularity of a song based on audio features and a variety of manual labels. The authors' conclusion is, however, that even state-of-the-art machine learning techniques fail to learn factors that determine a song's popularity, irrespective of whether they are trained on signal-based features or on high-level human annotations.

In [38] Schedl et al. propose several heuristics to determine which artists are popular within a certain genre. They relate occurrence counts of artist names on Web pages via an approach similar to *Google*'s backlink and forward link analysis [34]. The authors show that downranking factors for artist names equaling common speech terms improve accuracy when comparing the resulting rankings against a ground truth popularity categorization extracted from *allmusic.com* [3].
In [22] Grace et al. derive popularity rankings from user comments in the social network *MySpace* [32]. To this end, the authors apply various annotators to crawled *MySpace* artist pages in order to spot, for example, names of artists, albums, and tracks, sentiments, and spam. Subsequently, a data hypercube (OLAP cube) is used to represent structured and unstructured data, and to project the data to a popularity dimension. A user study showed that the list generated by this procedure was on average preferred to the Billboard charts.

Previous work that relates to the four approaches proposed here comprise the following.
Our heuristic that uses *page counts* returned by search engines builds upon work from [20, 39], where Web co-occurrences of artist names and terms specific to the music domain are used to categorize artists, a process also known as "autotagging" [13]. In [37] Schedl et al. propose a similar approach to estimate artist similarity. The authors suggest a simple probabilistic model that defines similarity between two artists a and b as the conditional probability of a to be mentioned on a Web page known to relate to b and vice versa. Accuracies of up to 85% were reported for genre classification.
To the best of our knowledge, *Twitter* [41] has not been scientifically investigated for music information extraction and retrieval yet. Although there do exist certain commercial services, such as *BigChampagne* [7] and *Band Metrics* [9], which seem to incorporate microblogging data into their artist and song rankings, no details on their approach are available. Furthermore, they strongly focus their services on the USA. A general study on the use of *Twitter* can be found in [24]. Java et al. report that *Twitter* is most popular in North America, Europe, and Asia (Japan), and that same language is an important factor for cross-connections ("followers" and "friends") over continents. The authors also distill certain categories of user intentions to microblog. Employing the *HITS* algorithm [25] on the network constructed by "friend"-relations, Java et al. derive user intentions from structural properties. They identified the following categories: information sharing, information seeking, and friendship-wise relationships. Analyzing the content of *Twitter* posts, the authors distill the following intentions: daily chatter, conversations, sharing information/URLs, and reporting news.
Using *Peer-to-Peer networks* as data source for music information retrieval, [8, 14, 31, 43] rely on data extracted from *OpenNap* to derive music similarity information. All of these papers seem to build upon the same data set, which comprises of metadata on shared content (approximately 3,000 shared music collections were analyzed). Logan et al. [31] compare similarities defined by artist co-occurrences in shared folders, by expert opinions from *allmusic.com*, by playlist co-occurrences from *Art of the Mix* [4], by data gathered from a Web survey, and by MFCC features [5]. To this end, they calculate a "ranking agreement score", i.e., the pairwise overlap between the N most similar artists according to each data source. The main findings are that the co-occurrence data from *OpenNap* and from *Art of the Mix* show a high degree of overlap, the experts from *allmusic.com* and the participants of the Web survey show a moderate agreement, and the signal-based MFCC measure had a rather low agreement with the music context-based data sources. More recently, in [40] Shavitt and Weinsberg mine the *Gnutella* file sharing network to derive artist and song similarities. The authors gathered metadata of shared music files from about one million *Gnutella* users in November 2007, which yielded information on half a million songs. Analyzing the 2-mode graph of users and songs revealed that most users share similar files. The authors further propose a method for artist recommendation based on the gathered data.
Taking a closer look at the data source of *music information systems*, which corresponds to the fourth approach, not only *last.fm* [28] provides popularity rankings via their API [29]. *Echonest* [15] offers a function to retrieve a ranking based on the so-called "hotttness" of an artist [17]. This ranking is based on editorial, social, and mainstream aspects [16]. However, this Web service does not provide country-specific information.

3. DETERMINING ARTIST POPULARITY ON THE COUNTRY LEVEL

We propose the following four heuristics to determine an artist's popularity in a certain country, and consequently create an artist popularity ranking. To this end, we first retrieve a list of 240 countries from *last.fm* [30], based on which the following approaches operate.

3.1 Search Engine Page Counts

This approach makes use of a search engine's number of indexed Web pages for a given query, a count usually referred to as *page count*. These page counts are, however, only rough estimates of the real number of available Web pages related to the query. Nevertheless, for the purpose of classifying music artists into genres [20, 37, 39] and for classifying general instances according to a given ontology as well as for learning sub- and superconcept relations [11, 12], this method yielded respectable results.

For the paper at hand, we queried the search engines *Google* [21] and *Exalead* [18], using their API or issuing HTTP requests. The page count values returned for all $\langle$artist, country$\rangle$ tuples were retrieved. To avoid excessive bandwidth consumption, we restrict the number of search results to be transmitted to the smallest value (this is usually one result). Since we are only interested in the page count estimates, this restriction effectively reduces network traffic without effecting the results.
The two main challenges of this approach are directing the search towards pages related to the music domain and alleviating the distortions caused by artist names that equal common speech words. We address these issues by using queries of the form

```
"artist name" "country name" music
```

and weighting the resulting page count values with a factor resembling the *inverse document frequency (idf)* [46]. The final ranking score is thus calculated according to Formula 1, where $pc_{c,a}$ is the page count value returned for the country-specific query for artist a and country c, N is the total number of countries for which data is available, and df_a is the number of countries in which artist a is known

according to the data source (i.e., the number of countries with $pc_{c,a} > 0$).

$$popularity\_pc_{c,a} = pc_{c,a} \cdot \log_2 \left(1 + \frac{N}{df_a}\right) \quad (1)$$

3.2 Twitter Posts

Many *Twitter* posts reveal information about what people are doing or thinking right now. We are interested in posts containing information about which music is currently being played by users in a given country. To accomplish this, we retrieve posts using the *Twitter* Search API [42]. The posts are then narrowed in two ways. First, we only search for posts containing the hashtag *#nowplaying*. This restriction is directly supported by the *Twitter* API. As a second restriction, the search is narrowed to a specific country. Not being aware of a more direct implementation for the second restriction, we search only for posts whose users are located within a certain radius around a GPS coordinate. More specifically, for a given country, we determine the coordinates of larger cities (with more than 100,000 inhabitants) and search for posts originating from a circle of 100 kilometers around the respective coordinates. The names of the cities are taken from *Wikipedia*, e.g., [45], and the coordinates are determined by using *Freebase* [19]. For each city location for which geolocation data is resolved successfully, all *Twitter* posts available through the *Twitter* API are retrieved, which yields a maximum of about 1,500 posts per city location.

One of the advantages of using this kind of data is certainly its recentness. Thus, the retrieved data may contain artists that do not appear in our list of most popular artists (cf. Section 4.1). A first look at the format of the texts reveals that automatic tokenization seems not easily to accomplish due to the large variation of wording and creative methods to use the available number of characters. We therefore opt to scan the retrieved texts for the artists contained in the artist list, and we count the number of their appearances for a given country c. This count equals the term frequency ($tf_{c,a}$) of a in an aggregated document comprising all posts gathered for cities in country c. Formula 2 gives the ranking score. The rightward term again represents an *idf*-factor that downranks artists that are popular everywhere, and thus not specific to country c. N is the total number of countries, and df_a is the number of aggregated country documents in which artist a occur.

$$popularity\_twi_{c,a} = tf_{c,a} \cdot \log_2 \left(1 + \frac{N}{df_a}\right) \quad (2)$$

3.3 Shared Folders in a P2P Network

Collecting shared folder data from *Gnutella* users is a two-staged-process. First, a *crawler* needs to discover the current network topology (which is very dynamic). Subsequently, a *browser* queries the active users for their shared folders data. The crawler treats the network as a graph, and performs a breadth-first exploration, where newly discovered nodes are enqueued in a list of un-crawled addresses. The crawler provides a list of active IP addresses to the browser, which sends *Gnutella* "Query" messages [1] to the clients. The clients reply with "QueryHit" messages, that lists their shared folder content. These messages are the basis for our P2P data set.

The system described above is a different system than the one used by Koenigstein and Shavitt in [26], which collected *Gnutella* search queries for song ranking. One advantage of a shared folder data set over queries is the availability of ID3 tags and hash keys, which simplifies the process of associating the digital content with a musical artist. However, when singles ranking is considered (as in [26]), queries tend to better reflect the changing popularity trends of pop songs over short time intervals. In this study, we associate artists with digital content by matching the artist names against the content of the ID3 tags. Occasionally, the content in ID3 tags is missing or misspelled. We therefore, match the artists names against the file names as well.

In order to build popularity charts for specific countries, one needs to resolve the geographical location of the users. The geo-identification is based on the IP addresses. First, we generate a list of all unique IP addresses in the data set (typically over a million). We resolve the geography of IP addresses using the commercial *IP2Location* [23] database. Each IP address is bounded with its country code, city name, and latitude-longitude values. This accurate geographical information pin points artists' fans and enables tracking spatial diffusion of artists popularity [27].

After the digital files are associated with artists names and geography, building popularity charts is straightforward. For each country, we aggregate the total number of digital content that is associated with each artist. Ranking is then performed according to frequency.

3.4 Last.fm Playcounts

We further estimate country-specific artist popularity based on the community of *last.fm* users. Despite the issues of *hacking and vandalism* as well as the *community bias* [36], which are inherent to collaborative music information systems, the playcounts of *last.fm* users can be expected to reflect to a certain extent which music is currently popular. We therefore gathered the top 400 listeners of each country at the end of 2009. We subsequently extracted the top-played artists for each of the resulting top-listeners-sets. [1] Aggregating the playcounts for each artist over a country's top listeners finally yielded a popularity ranking for the country under consideration.

4. EVALUATION

4.1 Test Set

We used *last.fm*'s Web API [29] to gather the most popular artists for each country of the world, as of November 2009. We then aggregated this data into a single list of 201,135 unique artist names.

4.2 Experiments

As we aim at assessing the pros and cons of the various approaches, without yet having an established ground truth for this kind of experiments, we choose to perform a pairwise comparison of the approaches. Each approach produces a ranked list of artists for the various countries. Expecting that the absolute numbers obtained by the various approaches are not immediately comparable, we compare the produced artist popularity rankings of two approaches

[1] In the meantime, *last.fm* has extended its API with a `Geo.getTopArtists` function, which can be used to directly retrieve the top-played artists among a certain country's users. Quick empirical comparisons showed that the implementation behind this function seems to resemble our approach.

A_j and A_k. This comparison is done separately for each country c. In the next subsections, we describe the applied data preprocessing steps and the used evaluation measure in detail.

4.2.1 Preprocessing

We start our analysis by processing the artist names in the artist popularity list for country c of each approach in a basic way (e.g., each artist name is represented in lower case, repeated whitespace characters are removed, and UTF-8-encoded characters are transformed to canonical ASCII representations).

Instead of using raw artist counts directly, we normalize them, attempting to avoid dominance of common-speech words, or globally popular artists whose popularity is not highly country specific. For each artist, the number of countries this artist appears in is counted. Each country-specific artist count $\mathrm{ac}_{c,a}$ is then normalized as indicated in Equation 1.

Artist names appearing in the two lists (given by the pair of approaches under investigation) are matched against each other, and only artists appearing in both lists are kept. Based on this data, we calculate the overlap between the rankings obtained with the two prediction approaches, as described next.

4.2.2 Evaluation Measures

The top-n rank overlap for country c between approaches A_j and A_k is calculated as

$$\mathrm{ro}_{c,A_j,A_k,n} = \frac{1}{n} \cdot \left|\{a | \max\left(r_{A_j,c,a}, r_{A_k,c,a}\right) \leq n\}\right| \quad (3)$$

where $r_{A_j,c,a}$ denotes the ranking of artist a in country c according to approach A_j, only considering the artists for which both approaches (A_j and A_k) yield a ranking score. In other words, the top-n rank overlap is the fraction of artists appearing within the top n ranked artists in both approaches. For example, if one artist is within the top-2 ranked artists of both approaches, the top-2 rank overlap is 0.5. Obviously, n can take values up to the number of artists $n_{\max,c}$ for which both approaches deliver rank data for country c, and the top-$n_{\max,c}$ rank overlap is always 1.

To obtain an overall measure for two approaches and a given country, we define the country-wise rank overlap as

$$\mathrm{cro}_{c,A_j,A_k} = \frac{1}{n_{\max,c}} \sum_{n=1}^{n_{\max,c}} \mathrm{ro}_{c,A_j,A_k,n} \quad (4)$$

which has a trivial (random) baseline of about 0.5 and a maximum value of 1.0 when both rankings are identical. The country-wise rank overlaps are further combined to obtain one overall scalar value for approaches A_j and A_k. To account for the different quantity of available information, we weight the overlap score of each country with the number of artists for which information is available. We define the overall overlap measure between approaches A_j and A_k as

$$\mathrm{ov}\left(A_j, A_k\right) = \frac{\sum_{c \in C} n_{\max,c} \cdot \mathrm{cro}_{c,A_j,A_k}}{\sum_{c \in C} n_{\max,c}} \quad (5)$$

The measure ov also has a trivial baseline of about 0.5 and a maximum value of 1.0.

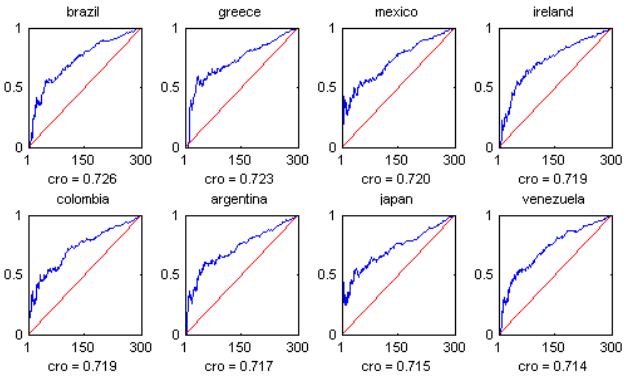


Figure 1. Top 8 countries for *pc_google* vs *p2p*.

To give an illustrative example, Figure 1 shows for the comparison of approach *pc_google* and *p2p* the 8 countries with highest ro value, as a chart from $1..n_{\max,c}$.

4.3 Results and Discussion

Each approach offers at least a slightly different view on reality since the data sources are of distinct nature. There is also no such thing as a "ground truth" for this task, as each data source (even "Billboard"-style charts) is biased, as elaborated below. Nevertheless, we would like to point out certain interesting observations.

Looking at Figure 2, the highest overlap score of 0.67 is found between *Google page counts* and *P2P*. One reason may be that the two sources have broadest coverage. Another explanation may be the time dependency. *Twitter* and *last.fm* are much more time dependent, whereas *P2P shared folders* and *amounts of Web pages* change much slower. In fact, the content of the data sources behind P2P networks and Web search engines, i.e., users' music collections and Web pages, respectively, is accumulated over years. Microblogging posts and *last.fm* data, in contrast, change much faster and are therefore more likely to reflect trends.

Second, the *page counts* approach using *Google* and the same approach using *Exalead* do not produce similar results, as we would have expected. In fact, the rankings reveal a non-significant overlap of 0.51. A possible explanation is that the two search engine providers may use very different page count estimation techniques.

Exalead shows the lowest overlap with other sources. Its highest overlap is realized, not surprisingly, with *Google* and with *P2P*, but it remains slightly above the baseline (0.53). An explanation for *Exalead*'s low overlap score becomes apparent when looking at Figure 3. *Exalead* has by far the highest number of matching artists, which may induce a high noisiness.

In terms of country coverage (cf. Figure 3), the *last.fm* and the *page counts* approaches offer data for nearly every country in the world.

To account for the different nature and scope of the proposed approaches (and underlying data sources), we compare them according to several aspects in Table 1, elaborating on specific advantages and disadvantages. One issue is that certain approaches are prone to a specific bias. For example, the average *last.fm* user does not represent the average music listener of a country, i.e., *last.fm* data is distorted by a "community bias". The same is true for *Twitter*, which is biased towards artists with very active fans. On the other hand, some very popular artists may have fans

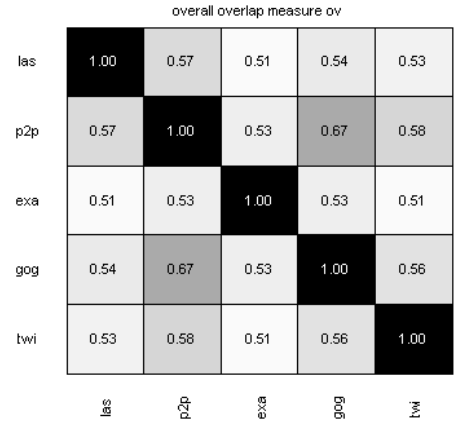


Figure 2. Overlap ov between each pair of approaches.

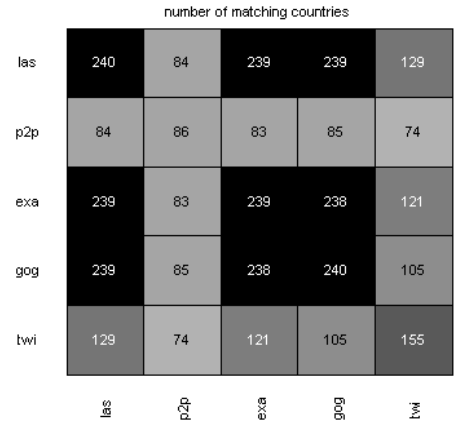


Figure 3. Number of countries with non-empty overlap.

that twitter to a much lower degree. This issue becomes especially apparent when thinking of live artists vs. dead ones: The live ones keep making new headlines, and probably also have many more active fans, while the dead ones have an inherent problem with this. Traditional charts are biased towards the data the music industry uses to derive them, usually record sales figures.

Another aspect according to which the approaches differ considerably is the availability of data. While *page count estimates* are available for all countries of the world, the *P2P* and *Twitter* approaches suffer from a very unbalanced coverage, strongly depending on the country under consideration. Also traditional music charts vary strongly between countries and continents with respect to availability. According to [44], only one country in Africa publishes official music charts, while this number amounts to 19 for Europe.

A big advantage of traditional charts is their virtual immunity against noise. *Page count estimates*, in contrast, are easily distorted by ambiguous artist or country names. *last.fm* data suffers from hacking and vandalism [10], as well as from unintentional input of wrong information and misspellings.

In the dimension of time dependence, the approaches can be categorized into "current" and "accumulating", depending on whether they reflect the instantaneous popularity, or a general, all-time popularity in that they accumulate popularity levels over time.

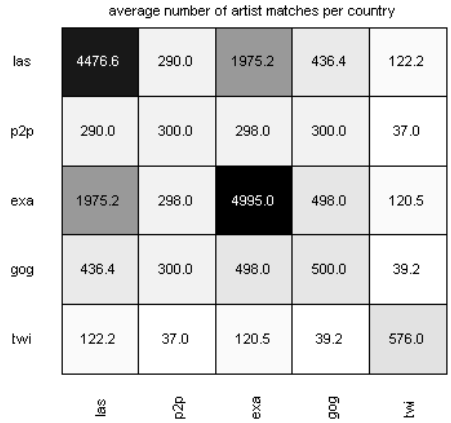


Figure 4. Average number of artists per country ($n_{\max,c}$).

5. CONCLUSIONS AND FUTURE WORK

We presented four approaches to determine country-specific artist popularity rankings based on different data sources (search engine's page counts, *Twitter* posts, shared folders in the *Gnutella* network, and playcounts of *last.fm* users). In the absence of a standardized ground truth, we performed pairwise comparison of the approaches and elaborated on particular advantages and disadvantages. Most approaches showed only weak overlaps, probably due to the different nature of their data sources. We found, however, a considerable overlap between *Google* page counts and P2P data, which is probably explained by the similar time scope the two data sources cover. As a general conclusion, we can state that artist popularity can be derived from various, quite inhomogeneous data sources. The remarkably weak overlap between most of them indicates that the quest for artist popularity is a multifaceted and challenging task, in particular in today's era of multi-channel music distribution. To derive one overall popularity measure, we will need to combine the different sources.

Future work will hence foremost aim at elaborating hybrid approaches that account for the different quantity and quality of information output by the four heuristics. We will also work on refining our approaches to capture artist popularity within certain genres, e.g., by incorporating methods similar to [38]. We will further look at the various processing steps in more detail. Most of the current implementations were created in an ad-hoc manner, and some of the choices might degrade the performance. For example, better string comparison algorithms may improve results for artists whose names may be spelled in various ways. Alternative ways of normalizing artist counts for the individual approaches are also likely to yield improvements.

6. ACKNOWLEDGMENTS

This research is supported by the *Austrian Fonds zur Förderung der Wissenschaftlichen Forschung* (FWF) under project numbers L511-N15 and Z159.

7. REFERENCES

[1] The Gnutella Protocol Specification v0.41. http://www9.limewire.com/developer/gnutella_protocol_0.4.pdf (access: March 2010).

Source/Aspect	Bias	Availability	Noisiness	Time Dependence
Page Counts	Web users	comprehensive	high	accumulating
Twitter	community	country-dependent	medium	current
P2P	community	country-dependent	low–medium	accumulating
Last.fm	community	high	medium–high	accumulating
Traditional Charts	music industry	country-dependent	low	current

Table 1. A comparison of different approaches according to various dimensions.

[2] Digital Music Report 2009. http://www.ifpi.org/content/library/DMR2009.pdf (access: May 2010), January 2009.

[3] http://www.allmusic.com (access: January 2010).

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

[5] Jean-Julien Aucouturier, François Pachet, and Mark Sandler. "The Way It Sounds": Timbre Models for Analysis and Retrieval of Music Signals. *IEEE Transactions on Multimedia*, 7(6):1028–1035, December 2005.

[6] http://en.wikipedia.org/wiki/Billboard_Hot_100 (access: May 2009).

[7] http://www.bigchampagne.com (access: May 2010).

[8] Adam Berenzweig, Beth Logan, Daniel P.W. Ellis, and Brian Whitman. A Large-Scale Evaluation of Acoustic and Subjective Music Similarity Measures. In *Proceedings of ISMIR*.

[9] www.bandmetrics.com (access: May 2010).

[10] Òscar Celma and Paul Lamere. ISMIR 2007 Tutorial: Music Recommendation.

[11] Philipp Cimiano, Siegfried Handschuh, and Steffen Staab. Towards the Self-Annotating Web. In *Proceedings of ACM WWW*, 2004.

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

[13] Douglas Eck, Thierry Bertin-Mahieux, and Paul Lamere. Autotagging Music Using Supervised Machine Learning. In *Proceedings of ISMIR*, 2007.

[14] Daniel P.W. Ellis, Brian Whitman, Adam Berenzweig, and Steve Lawrence. The Quest For Ground Truth in Musical Artist Similarity. In *Proceedings of ISMIR*, 2002.

[15] http://echonest.com (access: March 2010).

[16] http://developer.echonest.com/docs/method/get_hotttnesss (access: March 2010).

[17] http://developer.echonest.com/docs/method/get_top_hottt_artists (access: March 2010).

[18] http://www.exalead.com (access: February 2010).

[19] http://www.freebase.com (access: March 2010).

[20] Gijs Geleijnse and Jan Korst. Web-based Artist Categorization. In *Proceedings of ISMIR*, 2006.

[21] http://www.google.com (access: March 2010).

[22] Julia Grace, Daniel Gruhl, Kevin Haas, Meenakshi Nagarajan, Christine Robson, and Nachiketa Sahoo. Artist Ranking Through Analysis of On-line Community Comments. In *Proceedings of ACM WWW*, 2008.

[23] http://www.ip2location.com (access: March 2010).

[24] Akshay Java, Xiaodan Song, Tim Finin, and Belle Tseng. Why We Twitter: Understanding Microblogging Usage and Communities. In *Proceedings of WebKDD/SNA-KDD*, 2007.

[25] Jon M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM*, 46(5), 1999.

[26] Noam Koenigstein and Yuval Shavitt. Song Ranking Based on Piracy in Peer-to-Peer Networks. In *Proceedings of ISMIR*, 2009.

[27] Noam Koenigstein, Yuval Shavitt, and Tomer Tankel. Spotting Out Emerging Artists Using Geo-aware Analysis of P2P Query Strings. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008.

[28] http://last.fm (access: March 2010).

[29] http://last.fm/api (access: March 2010).

[30] http://www.last.fm/community/users (access: March 2010).

[31] Beth Logan, Daniel P.W. Ellis, and Adam Berenzweig. Toward Evaluation Techniques for Music Similarity. In *Proceedings of ACM SIGIR: Workshop on the Evaluation of Music Information Retrieval Systems*, 2003.

[32] http://www.myspace.com (access: November 2009).

[33] François Pachet and Pierre Roy. Hit Song Science is Not Yet a Science. In *Proceedings of ISMIR*, 2008.

[34] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank Citation Ranking: Bringing Order to the Web. In *Proceedings of ASIS*, 1998.

[35] Matei Ripeanu. Peer-to-Peer Architecture Case Study: Gnutella Network. In *Proceedings of IEEE Peer-to-Peer Computing*, 2001.

[36] Markus Schedl and Peter Knees. Context-based Music Similarity Estimation. In *Proceedings of LSAS*, 2009.

[37] Markus Schedl, Peter Knees, and Gerhard Widmer. A Web-Based Approach to Assessing Artist Similarity using Co-Occurrences. In *Proceedings of CBMI*, 2005.

[38] Markus Schedl, Peter Knees, and Gerhard Widmer. Investigating Web-Based Approaches to Revealing Prototypical Music Artists in Genre Taxonomies. In *Proceedings of ICDIM*, 2006.

[39] Markus Schedl, Tim Pohle, Peter Knees, and Gerhard Widmer. Assigning and Visualizing Music Genres by Web-based Co-Occurrence Analysis. In *Proceedings of ISMIR*, 2006.

[40] Yuval Shavitt and Udi Weinsberg. Songs Clustering Using Peer-to-Peer Co-occurrences. In *Proceedings of the IEEE ISM: International Workshop on Advances in Music Information Research (AdMIRe)*, San Diego, CA, USA, 2009.

[41] http://twitter.com (access: February 2010).

[42] http://apiwiki.twitter.com/Twitter-API-Documentation (access: March 2010).

[43] Brian Whitman and Steve Lawrence. Inferring Descriptions and Similarity for Music from Community Metadata. In *Proceedings of ICMC*, 2002.

[44] http://en.wikipedia.org/wiki/Music_charts (access: March 2010).

[45] http://en.wikipedia.org/wiki/List_of_towns_and_cities_with_100,000_or_more_inhabitants/country:_A-B (access: March 2010).

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

Markus Schedl

Leveraging Microblogs for Spatiotemporal Music Information Retrieval

Proceedings of the 35th European Conference on Information Retrieval (ECIR)

Moscow, Russia, March 2013

Leveraging Microblogs for Spatiotemporal Music Information Retrieval

Markus Schedl

Department of Computational Perception
Johannes Kepler University
Linz, Austria
http://www.cp.jku.at

Abstract. We present results of text data mining experiments for music retrieval, analyzing microblogs gathered from November 2011 to September 2012 to infer *music listening patterns* all around the world. We assess *relationships between particular music preferences and spatial properties*, such as month, weekday, and country, and the *temporal stability of listening activities*. The findings of our study will help improve music retrieval and recommendation systems in that it will allow to incorporate geospatial and cultural information into models for music retrieval, which has not been looked into before.

1 Introduction

Exploiting social media to enrich retrieval methods by including user-generated data is a quite recent strategy. In particular in Music Information Retrieval (MIR), work that leverages social media data is almost non-existent, except for publications that make use of the music service `Last.fm`[1]. On the other hand, retrieval methods that take into account cultural differences in the perception and consumption of music are highly desired [6]. Aiming to narrow this gap, we first determine music-related microblogs, extract from them information about location and music items (Section 2), annotate each item, and perform an analysis of the resulting annotated data collection that will eventually lead to *spatiotemporal music retrieval* methods. In particular, we investigate the *relationship between music preference and spatial properties* (Section 3) and the *temporal stability of listening patterns* (Section 4). Related work is sketched in Section 5; conclusions are drawn in Section 6. To foster reproducibility and further experimentation, the data collection named `MusicMicro 11.11-09.12` can be downloaded[2].

2 Determining Music Listening Patterns

We monitored the `Twitter` streaming API from November 2011 to September 2012, using the `Spritzer` feed[3] which contains a random selection of 1–2% of all tweets. To determine tweets related to music, we filtered the stream

[1] http://last.fm
[2] http://www.cp.jku.at/people/schedl/data/MusicMicro/musicmicro.html
[3] http://gnip.com/twitter/spritzer

with keywords typically used to communicate music listening activities, such as `#nowplaying`, `#np`, or `#itunes`. We further excluded tweets that did not contain a location. The resulting set still contains a lot of irrelevant microblogs[4]. For this reason, we extracted track and artist lists from `Musicbrainz`[5] and applied a pattern-based, multistage entity detection technique. Those tweets that could be identified as pointing to a music artist and/or song were retained. We furthermore excluded all tweets posted by users having the substring "radio" in their user names to suppress radio stations as they may distort the results. After these rigorous filtering steps, a total of 594,306 microblogs by 136,866 users, in which we identified 19,529 unique artist names, remained for subsequent investigation. Using the `Yahoo! PlaceFinder` API[6], we were able to connect the tweets to 20,722 different cities in 180 countries.

Since we ultimately aim at geospatial, semantic music search, we annotated each of the microblogs identified as described above with semantic tags. To this end, we gathered a set of 288 moods from `Allmusic`[7]. This set was then used to index collaborative tag lists extracted from `Last.fm`[8] for each artist, allowing to project the artist/tweet space to a semantic tag space. This projection (i) effectively reduces computational complexity and (ii) allows to retrieve music items by semantic labels, which is important when the user does not know the music item she is searching for.

3 Relation: Listening Preferences – Spatial Properties

To obtain the overall distribution of music listening activity, we compute a normalized, worldwide *tag distribution vector* $\boldsymbol{T}$ as $\frac{\sum_{c \in C} \sum_{u \in U(c)} \frac{\boldsymbol{T(c,u)}}{|A(u)|}}{\sum_{c \in C} |U(c)|}$, where C is the set of countries, $U(c)$ are the unique users in country c, and $\boldsymbol{T}(c,u)$ is the tag distribution vector of user u in country c, i.e., a vector containing aggregated tag occurrence counts for $A(u)$ (multiset of artists user u has listened to). The normalized tag distribution vector $\boldsymbol{T(c)}$ of a particular country c is likewise defined as $\boldsymbol{T(c)} = \sum_{u \in U(c)} \frac{\boldsymbol{T(c,u)}}{|A(u)|}$.

Figure 1 depicts the most frequent 20 tags and the five countries with highest tweeting activities. We see that preference for particular *music mood varies considerably between countries*, a fact that will allow future music retrieval approaches to personalize results by incorporating spatial information. The bar chart shows the relative difference between $\boldsymbol{T(c)}$ and $\boldsymbol{T}$ as the normalized Manhattan distance: $\frac{L_1(\boldsymbol{T(c)},\boldsymbol{T})}{\boldsymbol{T}}$. For instance, "lyrical" music is listened to in the USA 116% more frequently than the worldwide average suggests. "Smooth" music is listened to in Brazil 53% less frequently than the worldwide average.

Quantitative analysis of listening correlation between countries (independent of time), similar to the approach presented in the previous section, shows indeed a relatively low mean correlation and a high standard deviation: $\overline{\rho} = 0.7382\ (\pm 0.2012)$, $\min(\rho) = -0.0238$, $\max(\rho) = 1.0$.

[4] For instance, `#np` is also used to indicate playing video games.
[5] http://musicbrainz.org
[6] http://developer.yahoo.com/geo/placefinder
[7] http://allmusic.com
[8] http://last.fm/api

4 Temporal Stability of Listening Activities

To investigate whether listening patterns are consistent over time (independent of the country), we compute Pearson's correlation coefficient between the normalized tag vectors for each pair of months: $\rho\left(\overline{\boldsymbol{T}(m_i)}, \overline{\boldsymbol{T}(m_j)}\right)$. The mean ρ value over all pairs of months is $\overline{\rho} = 0.9974\,(\pm 0.0021)$; the maximum is 0.998 (January vs. February 2012); the minimum is 0.9902 (November 2011 vs. September 2012). We thus conclude that *music listening patterns are highly independent of month*.

Analyzing analogously the correlation between tag vectors aggregated at the level of weekdays, we find that listening patterns are highly correlated between weekdays. In fact, the mean correlation between aggregated tag vectors among workdays only and among weekends only is 0.9999. The corresponding mean correlation between workdays and weekends is 0.9993. Nevertheless, due to the large sample size, the difference between the two correlation values is significant, according to Fisher's r-to-z transformation. There is hence a *significant difference in listening behavior between workdays and weekends*, which may be explained by different music preferences during working and partying hours.

5 Related Work

There meanwhile exists extensive literate on the topics of social media mining (SMM) and –retrieval. For instance, Alhadi et al. recently presented an approach to predict interesting tweets, based on the retweet activity of users [1]. A comprehensive overview of related methods can be found in [4].

When it comes to geospatial analysis for music retrieval, in contrast, only very recently researchers have looked into culture-specific music creation and listening [6]. Serra's work so far focused on musical properties of non-western music. Bridging SMM and MIR research, Zangerle et al. present an approach to music recommendation based on microblog co-occurrences of artist and track names [7]. Schedl and Hauger present an approach to extract music genre patterns for different regions of the world [5] and a user interface to explore these patterns [3]. Unlike in the paper at hand, previous work typically focused on the aspect of music genre, which is known to be an ill-defined concept [2]; whereas we present a more general approach and further consider spatiotemporal aspects.

6 Conclusion and Outlook

Based on a microblog collection covering 11 months, we presented an approach to annotate tweets with music artist/track names, semantic tags, and geographic data. We then use these annotations to *infer music listening activities* and relate them to spatiotemporal properties. We found that *music listening is independent of month*. There is a statistically significant *difference between workdays and weekends and between countries* (irrespective of time), though.

As part of future work, we will exploit more specific information – tracks to describe listening activities and cities to describe corresponding locations. Incorporating the findings of this study, we will elaborate music retrieval systems providing serendipitous experiences [8]. We will further look into domains other than music, e.g., movies, politicians, or shares.

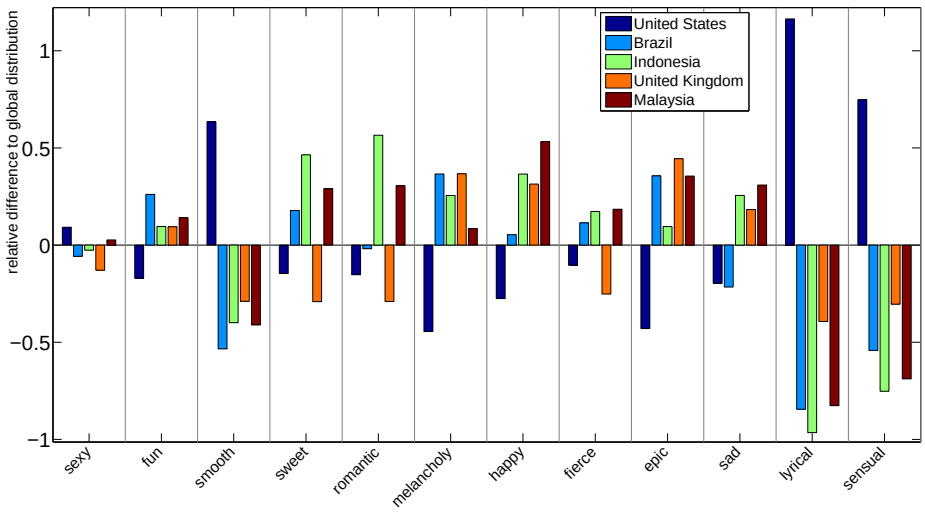


Fig. 1. Music mood distribution for top countries and top tags.

7 Acknowledgments

This research is supported by the Austrian Science Funds (FWF): P22856-N23.

References

1. A. C. Alhadi, T. Gottron, J. Kunegis, and N. Naveed. LiveTweet: Monitoring and Predicting Interesting Microblog Posts. In *Proc. ECIR*, Apr 2012.
2. J.-J. Aucouturier and F. Pachet. Representing Musical Genre: A State of the Art. *Journal of New Music Research*, 32(1):83–93, 2003.
3. D. Hauger and M. Schedl. Exploring Geospatial Music Listening Patterns in Microblog Data. In *Proc. AMR*, Oct 2012.
4. N. Ramzan, R. van Zwol, J.-S. Lee, K. Clüver, and X.-S. Hua, editors. *Social Media Retrieval*. Springer, Nov 2012.
5. M. Schedl and D. Hauger. Mining Microblogs to Infer Music Artist Similarity and Cultural Listening Patterns. In *Proc. WWW Workshop: AdMIRe*, Apr 2012.
6. X. Serra. Data Gathering for a Culture Specific Approach in MIR. In *Proc. WWW Workshop: AdMIRe*, Apr 2012.
7. E. Zangerle, W. Gassler, and G. Specht. Exploiting Twitter's Collective Knowledge for Music Recommendations. In *Proc. WWW Workshop: #MSM*, Apr 2012.
8. Y. C. Zhang, D. O. Seaghdha, D. Quercia, and T. Jambor. Auralist: Introducing Serendipity into Music Recommendation. In *Proc. WSDM*, Feb 2012.

Markus Schedl, Tim Pohle, Peter Knees, Gerhard Widmer

Exploring the Music Similarity Space on the Web

ACM Transactions on Information Systems, 29(3), July 2011

Exploring the Music Similarity Space on the Web

MARKUS SCHEDL, TIM POHLE, PETER KNEES, and GERHARD WIDMER,
Johannes Kepler University

14

This article comprehensively addresses the problem of similarity measurement between music artists via text-based features extracted from Web pages. To this end, we present a thorough evaluation of different term-weighting strategies, normalization methods, aggregation functions, and similarity measurement techniques. In large-scale genre classification experiments carried out on real-world artist collections, we analyze several thousand combinations of settings/parameters that influence the similarity calculation process, and investigate in which way they impact the quality of the similarity estimates. Accurate similarity measures for music are vital for many applications, such as automated playlist generation, music recommender systems, music information systems, or intelligent user interfaces to access music collections by means beyond text-based browsing. Therefore, by exhaustively analyzing the potential of text-based features derived from artist-related Web pages, this article constitutes an important contribution to context-based music information research.

Categories and Subject Descriptors: H.4 [**Information Systems**]: Information Systems Applications; H.3 [**Information Systems**]: Information Storage and Retrieval

General Terms: Algorithms, Experimentation, Measurement

Additional Key Words and Phrases: Music information retrieval, Web content mining, term space, evaluation

ACM Reference Format:
Schedl, M., Pohle, T., Knees, P., and Widmer, G. 2011. Exploring the music similarity space on the Web. ACM Trans. Inf. Syst. 29, 3, Article 14 (July 2011), 24 pages.
DOI = 10.1145/1993036.1993038 http://doi.acm.org/10.1145/1993036.1993038

1. INTRODUCTION

Music Information Retrieval (MIR) is a steadily growing field of research. Although early work on how to apply information retrieval (IR) techniques to music dates back to the 1960s [Kassler 1966], MIR's broad emergence as a scientific discipline originates in the late 1990s, when computational power, network bandwidth and storage capabilities reached levels that made feasible signal-based processing and analysis of digital music data. As pointed out in Downie [2003], MIR is a multidisciplinary research endeavor that strives to develop innovative content-based searching schemes, novel interfaces, and evolving networked delivery mechanisms in an effort to make the world's vast amount of music accessible to all. MIR hence comprises actions, methods, and procedures for recovering stored data to provide information on music [Fingerhut 2004]

This research is supported by the Austrian Science Funds under project numbers L511-N15, P22856-N23, and Z159.
Authors' address: Johannes Kepler University, Department of Computational Perception, Altenberger Straße 69, 4040 Austria; email: markus.schedl@jku.at.

DOI 10.1145/1993036.1993038 http://doi.acm.org/10.1145/1993036.1993038

and 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) [Schedl 2008].

These definitions of MIR already indicate that it is a highly dynamic and multidisciplinary field of research that relates to various other research disciplines. Narrowing the focus to information extraction (IE) and information representation related to music, we can distinguish three broad categories of strategies in terms of the underlying data source, namely *music content*-based, *music context*-based and *user context*-based approaches. Feature vectors describing aspects from one or more of these three categories can be constructed, and similarity measures can be applied to the resulting vectors of two pieces of music or two music artists.[1] Elaborating such musical *similarity measures* that are capable of capturing aspects that relate to real, perceived similarity is one of the main challenges in MIR. At this point the reader may ask why music similarity is such an important concept. First, music similarity measures can help to understand why two music pieces or artists are perceived alike by the listener. In fact, a listener may state that two songs resemble each other, but cannot tell why they are similar. In this case, computational music similarity measures could give an explanation. Second, similarity measures are of particular importance in the music domain, because, unlike in image retrieval, where the viewer can mentally process the main content of an image within 150 msecs [Thorpe et al. 1996], a piece of music requires a much longer processing time by the auditory system. Music similarity measures are hence important to guide the user in efficiently retrieving a desired piece of music. Consequently, they are a key ingredient of various music-related applications. Examples are systems to automatically generate playlists [Aucouturier and Pachet 2002; Pohle et al. 2007c], music recommender systems [Celma and Lamere 2007; Zadel and Fujinaga 2004], music information systems [Schedl 2008], semantic music search engines [Knees et al. 2007], and intelligent user interfaces [Pampalk and Goto 2007; Knees et al. 2007] to access music collections by means more sophisticated than the textual browsing facilities (`artist-album-track` hierarchy) traditionally offered.

Methods to derive content-based features [Casey et al. 2008] extract information from a data source that represents the content of a piece of music. Most frequently, this is some manifestation of a song's audio signal, for instance, an mp3 file. Such content-based methods allow to model certain aspects of music that relate to acoustic properties. They are capable of describing, for example, the timbre ("sound") of a piece of music [Aucouturier and Pachet 2004] or its rhythmical structure [Pampalk et al. 2002; Schedl et al. 2005]. Recent work addresses more specific high-level aspects, such as melodiousness and "percussiveness," that is, the strength of percussive sounds in the signal [Pampalk 2006; Pohle 2009; Seyerlehner et al. 2007].

Music content-based approaches, however, fall short of describing some aspects that are important to the perception and understanding of music, but are not encoded in the audio signal. For example, an artist's geographic and cultural context, the political background, or the meaning of the song lyrics are likely to influence how his or her music is perceived, but cannot be detected from the music content. Therefore, an analysis of the music context [Schedl and Knees 2009] is necessary if we aim at distilling such factors. Among many other data sources, some of which will be presented as part of related work, an obvious source for contextual data is the World Wide Web. First steps towards using Web pages to derive term feature vectors for the purpose of artist similarity calculation were undertaken in Cohen and Fan [2000a], Whitman and Lawrence

[1]For reasons of simplicity, we use the term "artist" in the following to denote individual performers as well as bands performing music.

[2002], and Knees et al. [2004]. In this work the authors usually select a specific variant of the $tf \cdot idf$ term weighting measure [Baeza-Yates and Ribeiro-Neto 1999] and apply it to Web pages retrieved for music artists. The individual choices involved in selecting a specific $tf \cdot idf$ variant and similarity function, however, do not seem to be the result of detailed assessments. They rather resemble common variants that are known to yield good results in IR tasks. Whether these variants are also suited to describe music artists via term profiles and subsequently estimate similarities between them is seldom assessed comprehensively in the literature on text-based music information extraction.

Addressing this lack of investigation, we present the first comprehensive study on Web-based music similarity estimation. Our work is inspired by Zobel and Moffat [1998], where the authors thoroughly evaluate various decisions involved in constructing text-based feature vectors for IR purposes, for instance, term frequency, term weights (idf), and normalization approaches. They analyze the influence of these decisions on retrieval behavior. Similarly, we present a large-scale study on the influence on similarity estimation of a multitude of decisions, using real-world data collections. To this end, we analyze several thousand different combinations of the following single aspects:

—term frequency;
—inverse document frequency;
—virtual document modeling;
—normalization with respect to page length;
—similarity function.

The *term frequency* $r_{d,t}$ of a term t in a document d estimates the importance t has for document (related to artist) d. The *inverse document frequency* w_t estimates the overall importance of term t in the whole corpus and is commonly used to weight the $r_{d,t}$ factor, that is, downweight terms that are important for many documents and hence less discriminative for d. *Virtual document modeling* relates to the way individual documents retrieved for the same artist are aggregated. We further assess the impact of *normalization* with respect to length of individual Web pages. Different *similarity functions* S_{d_1,d_2} estimate the proximity between the term vectors of two documents or artists d_1 and d_2.

For reasons of completeness, let us state that the third category of MIR-related data sources, the user context, is not directly related to properties of music pieces or artists. It rather comprises external factors that influence how a listener perceives music. Examples for such aspects are the situation in which the listener consumes music (active vs. passive listening, romantic dinner, relaxed evening after a stressful day, preparing to go out on a Saturday night, playing music him/herself in a band), the listener's mood, his or her location, the used listening device (PC, stereo, cell phone, mobile music player), and the listener's social context (friends, peer groups, neighbors, listener's role in the context). In Göker and Myrhaug [2002] a general categorization of user context aspects is presented. However, user context aspects will not be discussed in detail in this contribution.

The remainder of this article is organized as follows. Section 2 outlines the context of this work by conducting a literature review on music context-based similarity estimation. Section 3 then discusses common approaches to extracting music-related information from the Web and details the specific approach we employed. An analysis and discussion of different decisions in the artist description, term weighting, and similarity measurement process can be found in Section 4. Finally, conclusions are drawn in Section 5.

2. BACKGROUND AND RELATED LITERATURE

Estimating similarities between music artists can be performed based on three categories of data sources: music content, music context, and user context. Since we investigate the use of Web pages to derive similarities in this article, we will review related work on Web-based music information extraction methods. Since Web pages reflect human knowledge and opinions, such methods hence fall into the category of music context-based approaches.

2.1. Explicit Similarity Data Collection

The most straightforward way to collect information about artist similarity, or related information such as genre, is by letting people explicitly deliver it. For example, Berenzweig et al. [2003] present a Web-based user survey asking people about their similarity judgments in a set of 400 artists.

Another source of musical knowledge is expert opinions. The music information system allmusic.com, for example, provides for each artist a list of similar artists and a list of genres the artist is assigned to. In a number of publications, such expert opinions have been used as ground truth to evaluate automated approaches [Berenzweig et al. 2003; Ellis 2002].

In recent years, *tagging* has become more popular. For example, the online music platform last.fm lets users assign tags to pieces of music, or music artists. This tag data is made available via an API. Another approach is to collect tags in the form of a game [Law et al. 2007; Mandel and Ellis 2007; Turnbull et al. 2007; Turnbull et al. 2008]. The basic principle of the tagging game [Ahn and Dabbish 2004] is to present the same item (which is a song in this case) to two different players, asking them to provide tags. Points are rewarded when both users provide matching tags. Tags that are proposed multiple times are taken as valid annotations for the item.

2.2. User Collections and Playlists

While explicitly asking people to provide similarity information is usually a source of high-quality data, it is also a very time-consuming task. A less time-consuming alternative to obtain certain types of information about music is to analyze user data, such as which music users have in their music collection, and how often they listen to which artists or songs. For example, Whitman and Lawrence [2002] calculate artist similarity based on cooccurring artists shared by users of the Peer-to-Peer (P2P) network OpenNap.[2] In a more recent work Shavitt and Weinsberg [2009] derive similarity information at the artist level and at the song level from the Gnutella P2P file sharing network. Shavitt and Weinsberg collected metadata of shared files from more than 1.2 million Gnutella users in November 2007, restricting their search to music files (.mp3 and .wav). The crawl yielded a data set of 530,000 songs. They used the data for song clustering and artist recommendation.

Alternatively, music playlists can be analyzed for track or artist cooccurrence patterns [Logan et al. 2003; Stenzel and Kamps 2005]. Playlists created by human users can be obtained, for example, from artofthemix.org or mixtape.me. Exploiting playlists to derive artist similarity information is performed in Baccigalupo et al. [2008], where the authors analyzed cooccurrences of artists in playlists shared by members of a Web community. The authors looked at more than 1 million playlists made publicly available by MusicStrands,[3] a Web service (no longer in operation) that allowed users to share playlists. The authors extracted the 4,000 most popular artists from the full playlist

[2]http://opennap.sourceforge.net.
[3]http://music.strands.com.

set, measuring the popularity as the number of playlists in which each artist occurs. They further take into account that two artists that consecutively occur in a playlist are probably more similar than two artists that occur farther away in a playlist. The authors use this data to define fuzzy genre membership of artists.

2.3. Song Lyrics

The lyrics of a song represent an important aspect of the semantics of music since they are typically closely tied to the artist or the performer by revealing, for example, cultural background, political orientation, or style of music (use of a specific vocabulary in certain music styles).

Logan et al. [2004] use lyrics of songs by 399 artists to determine artist similarity. To this end, in a first step, Probabilistic Latent Semantic Analysis [Hofmann 1999] is applied to a collection of over 40,000 song lyrics to extract N topics typical to lyrics. In a second step, all lyrics by an artist are processed using each of the extracted topic models to create N-dimensional vectors of which each dimension gives the probability of the artist's tracks to belong to the corresponding topic. Artist vectors are then compared by calculating the L_1 distance (also known as Manhattan distance). Evaluation is performed against human similarity judgments, that is, the "survey" data for the uspop2002 set [Berenzweig et al. 2003]. Logan et al.'s approach does not reach performance levels similar to those obtained via acoustic features (irrespective of the chosen N, the usage of stemming, or the filtering of lyrics-specific stopwords). However, as lyrics-based and audio-based approaches make different errors, a combination of both is suggested.

Mahedero et al. [2005] demonstrate the usefulness of lyrics for similarity measurement, among other tasks. A standard $tf \cdot idf$ measure with cosine distance is proposed as initial step. Using this information, a song's representation is obtained by concatenating distances to all songs in the collection into a new vector. These representations are then compared using an unspecified algorithm. Exploratory experiments indicate some potential for cover version identification and plagiarism detection.

The goal of Laurier et al. [2008] is classification of songs into four mood categories by means of lyrics and content analysis. For lyrics, the $tf \cdot idf$ measure with cosine distance is incorporated. Optionally, also Latent Semantic Analysis [Deerwester et al. 1990] is applied to the $tf \cdot idf$ vectors (achieving best results when projecting vectors down to 30 dimensions). In both cases, a 10-fold cross validation with k-nearest neighbor (k-NN) classification yielded accuracies slightly above 60%. Audio-based features performed better compared to lyrics features, however, a combination of both yielded best results.

Hu et al. [2009] experiment with $tf \cdot idf$, tf, and Boolean vectors and investigate the impact of stemming, part-of-speech tagging, and function words for soft-categorization into 18 mood clusters. Best results are achieved with $tf \cdot idf$ weights on stemmed terms. An interesting result is that in this scenario, lyrics-based features alone can outperform audio-based features.

3. WEB PAGE ANALYSIS

This section reviews a number of ways to obtain data relevant for music retrieval from the Web. Furthermore, our specific Web-based approach to automatically deriving information about similarity of music artists is presented. By querying a search engine, a number of Web pages is collected for each artist, and the subsequent use of text mining techniques allows for computing a similarity score between two artists.

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.

The process of obtaining music-related metadata from the Web by using text information retrieval techniques can be divided into three stages: data acquisition, data analysis and usage, which are discussed in the following.

3.1. Data Acquisition

The first step towards building a Web-based music similarity measure consists of identifying Web pages related to the music domain, for example, fan pages, biographies, album reviews, track lists, or sale offers for albums or songs. This Webpage selection can be carried out either by using a *focused crawler* [Chakrabarti et al. 1999] or by relying on Web search engines. Using a specialized focused crawler has the potential of yielding better pages as it intends to effectively confine the crawl to the music domain. However, since it involves various complex components (e.g., link analyzer and classifier), computational performance is generally limited. Issuing queries to a Web search engine to obtain related pages, in contrast, is fast and easy. On the other hand, the number of allowed automatically sent queries is usually limited and the ranking algorithm applied by the search engine is in most cases a well-kept secret.

Automatically querying a Web search engine to determine pages related to a specific topic is a common and intuitive task, which is therefore frequently performed in IE research. Examples in the music domain can be found in Whitman and Lawrence [2002] and Geleijnse and Korst [2006], whereas Cimiano et al. [2004], Cimiano and Staab [2004], and Knees et al. [2007] apply this technique in a more general context. Although this approach seems to be straightforward, it is prone to a major type of error: When searching for artist names that equal common speech words, usually a lot of irrelevant pages are returned.[4] Hence, the main challenge is to restrict the search results to pages related to the desired artist. This problem is commonly addressed by enhancing the search query for the artist name with additional keywords. In the context of music information research, Whitman and Lawrence [2002] proposed to confine the search by the keywords "music" and "review" in order to direct it towards album reviews. The resulting query scheme has successfully been applied in genre classification tasks [Knees et al. 2004]. Later research has shown that other keywords seem to yield more accurate results, depending on the task. For example, when aiming at determining band members, the query schemes "*artist*" `music` and "*artist*" `music members` were more successful [Schedl and Widmer 2007]. To gather general, music-related Web pages, the scheme "*artist*" `music` usually represents a good trade-off between coverage and false positives. Hence, we used it for the article at hand. It has to be borne in mind, however, that these settings are not suited for multilingual pages and artists for which no English content is available on the Web. Varying the language of the additional keywords (e.g., `music`, `Musik`, `musique`, `musica`) may resolve this issue, but at the price of considerably increasing the number of queries issued to the search engine. For almost all artists in our test collections, the number of available Web pages is well above the number of actually retrieved ones. Restricting the search to English keywords therefore does not impose any limitations concerning the quantity of artist-related pages analyzed. However, one should be aware that, in general, restricting the search space to English pages might yield undiscovered pages that are nevertheless relevant to the artist.

We first query Google's search engine to retrieve up to the top 100 URLs for each artist in the collection. We then fetch the Web content available at these URLs using an optimized fetcher featuring load balancing, which we implemented in Java.

[4]In the music domain typical artists that cause such problems are *Bush*, *Prince*, *Kiss*, and *Porn*.

Subsequently, we create a *full inverted index*, also known as *world-level index* [Zobel and Moffat 2006], using a modified version of the open-source indexer Lucene Java.[5]

3.2. Text Analysis and Processing

From the kind of data acquired in the previous step, Whitman and Lawrence [2002] extract *unigrams* (single words occurring in the texts), *bigrams* (pairs of words following each other in the texts), words that are likely to be *adjectives* (by applying a part-of-speech (POS) tagger), and noun phrases. Each of these forms a possible basis for a vector space, where each term (e.g., bigram) is one dimension. In Pampalk et al. [2005], as an alternative to generating the space out of the retrieved documents, a predefined dictionary of words is used that are meaningful in the music domain. To cope with different forms of the same word, a stemming algorithm can be used [Celma et al. 2006; Schnitzer et al. 2007] at the expense of potentially introducing ambiguities. In many cases, words that are very frequent (such as *the, I*) and thus are assumed to not carry a meaning in the particular domain are removed by using *stopword lists*.

The actual value $w_{d,t}$ assigned to an artist d in each dimension of the term space is computed from the frequency with which the term t occurs in documents related to this artist (*term frequency*, $r_{d,t}$), and typically is normalized by the count of the number of documents in which the term occurs (*document frequency*, w_t). The resulting vector is generally referred to as $tf \cdot idf$ vector. The basic intention of the tf factor is to assign higher weights to terms that occur more frequently on pages retrieved for artist d, whereas the inverse document frequency idf factor downweights terms that often occur in the whole corpus for different artists and therefore are not specific to artist d. Most formulations of idf apply the logarithm to the raw document frequency values to particularly suppress terms with very high df values (cf. Table III).

The preceding procedure is used to create a $tf \cdot idf$ vector space from the retrieved documents. However, there exist scenarios where other representations are used. For example, for the task of artist recommendation, in Cohen and Fan [2000] lists of artists are extracted from Web pages to eventually construct pseudo-users for a collaborative filtering approach. In Pachet et al. [2001], texts are analyzed for the occurrence of track and artist names to facilitate cooccurrence and correlation analysis for similarity computation. Schedl et al. [2007] combine named entity detection and a rule-based IE approach to derive band memberships. Approaches to predict the geographic origin of an artist are presented in Govaerts and Duval [2009] and Schedl et al. [2010].

An alternative term weighting scheme is the *BM25* function that is used in the Okapi framework for text-based probabilistic retrieval [Robertson et al. 1995; Robertson et al. 1999]. This model assumes a priori knowledge on topics from which different queries are derived. Moreover, based on information about which documents are relevant for a specific topic and which are not, the term weighting function can be tuned to the corpus under consideration. Since *BM25* is a well-established term-ranking method, we included it in the experiments. However, it has to be noted that in our case, we cannot assume any a priori classification, neither on the level of Web pages, nor on the artist level. On the Webpage level, manually classifying hundreds of thousands of Web pages would be too labor-intensive. On the artist level, we could obviously group the artists (or more precisely, the retrieved Web pages of the artists) according to a genre taxonomy and optimize *BM25* correspondingly. However, we believe that this is not justifiable for two reasons: First, for arbitrary music collections, we cannot assume to have genre information given. Second, using genre information would obviously bias the evaluation results for the genre classification experiments as the other term

[5]http://lucene.apache.org.

Table I. Denominations for Terms Commonly Used in Text Information Retrieval

$\mathcal{D}$	set of documents
N	number of documents
$f_{d,t}$	number of occurrences of term t in document d
f_t	number of documents containing term t
F_t	total number of occurrences of t in the collection
$\mathcal{T}_d$	the set of distinct terms in document d
f_d^m	the largest $f_{d,t}$ of all terms t in d
f^m	the largest f_t in the collection
$r_{d,t}$	term frequency; see Table II
w_t	inverse document frequency; see Table III
W_d	document length of d

weighting measures do not incorporate such a priori knowledge. Thus, *BM25* would be unjustifiably favored.

For our experiments, we therefore used a simpler *BM25* formulation as the one proposed in Robertson et al. [1999], cf. Section 4.1.4.

3.3. Usage

In a number of cases, data usage is tightly coupled with the previous steps (i.e., data retrieval and processing are chosen and designed with a particular application in mind). However, some of the data representations can be used for a variety of applications. Most notably, if a similarity function can be built on the extracted data, potential data usages include clustering, classification, and recommendation. Besides genre classification [Knees et al. 2004], it has been proposed to classify record reviews into classes of "like" and "dislike" [Hu et al. 2005], which eventually could be used to create recommendation systems with improved recommendation performance, for instance, by using only those record reviews that are known to be in line with the user's taste. Another application scenario is a user interface where the user can browse an artist collection via topics automatically derived from $tf \cdot idf$ vectors [Pohle et al. 2007a, 2007b].

In our large-scale analysis of Web-based music artist similarity measures, we derive and evaluate different variants of vector space representations as described in Section 4.

3.4. Similarity Estimation Approaches in Previous Work

A look into the literature reveals that there exist different ways to transform Web pages to a vector of term weights for artists. For example, differences lie in the way basic concepts of text information retrieval, most notably the concept of a *document*, are transferred to music artists who are represented by a number of Web pages. In the following, we use the denominations listed in Table I to refer to various terms of this domain.

Whitman and Lawrence [2002] and Whitman [2005] treat each artist as one document for calculating the document frequency (f_t), while term frequency ($f_{d,t}$) is the percentage of Web pages containing the term. Both f_t and $f_{d,t}$ are normalized, being considered a probability distribution (f_t are normalized after summing up over all artists, while $f_{d,t}$ are normalized for each artist), then $tf \cdot idf$ is computed and normalized for each artist individually in the range 0..1. Optionally, very frequent and very infrequent terms are downweighted by a Gaussian function. The similarity of two artist vectors is calculated by summing up the weights of terms occurring for both artists.

In Baumann and Hummel [2003] and Knees et al. [2004], $f_{d,t}$ is the number of occurrences of term t on the Web pages related to an artist d, and the document

frequency f_t is the number of Web pages the term occurs on (not the number of artists for which the term occurs).

Baumann and Hummel [2003] and Knees et al. [2004] differ in the way N is defined and the $tf \cdot idf$ vector is calculated, while both use the cosine similarity measure to compare artist vectors. Baumann and Hummel [2003] define N as the size of "the entire artist collection", and $tf \cdot idf$ is computed as

$$w_{d,t} = f_{d,t} \cdot \log\left(\frac{N}{f_t}\right). \tag{1}$$

In Knees et al. [2004], N is the total number of pages that were retrieved. For $tf \cdot idf$ computation the following variant is used:

$$w_{d,t} = \begin{cases} (1 + \log_2 f_{d,t}) \log_2 \frac{N}{f_t} & \text{if } f_{d,t} > 0, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

As motivated by these examples, there is no standard way to calculate $tf \cdot idf$ vectors from retrieved Web pages, and it is unclear which way to calculate it is preferable. In the next section, a number of variants to obtain $tf \cdot idf$ vectors (and how to compare them) is evaluated to gain some insight into this question.

4. EVALUATING VARIANTS OF TERM WEIGHTING, NORMALIZATION, AND DISTANCE MEASURES

As outlined above, it is a common technique to obtain descriptions of artists by analyzing the text of Web pages returned by a search engine queried with the artist name (and additional query terms to narrow the search to pages more relevant for the domain of music). This "search engine" approach has several advantages. First, the obtained data can be used in different ways (similarity computation [Knees et al. 2004], tagging artists [Schedl and Pohle 2010], categorizing artists [Geleijnse and Korst 2006], or deriving specific information [Schedl et al. 2007]). Second, this approach does not crucially depend on the availability of a specific online platform providing the particular type of data sought. Also current trends in music (e.g., emerging genres) are likely to be reflected in the returned pages quickly. Furthermore, future advances in indexing and search engine technology (finding more relevant pages related to an artist) can be expected to enhance the results.

In this section, we present the evaluation experiments conducted to assess different algorithm variants for calculating artist similarity based on term feature vectors. To assess the quality of the results, we perform genre classification experiments. Even though musical genre is an ill-defined concept and genre taxonomies tend to be highly inconsistent [Pachet and Cazaly 2000], we unfortunately do not have access to reliable and comprehensive similarity data, against which we could perform comparison. We therefore opted for a genre classification task that serves as proxy for artist similarity. We used a k-NN classifier (leave-one-out), and we investigated classification accuracy for different values of k. The assumption underlying the genre classification setting is that similar artists are assigned to the same genre. In leave-one-out classification, the training set consists of all artists except the one to be classified. For each seed artist a, it is tested whether the k closest neighbors' genre labels match a's genre label (where closeness is measured by the similarity algorithm under evaluation). The majority of a's closest neighbors' genre labels is used to classify a. Classification accuracy is computed as arithmetic mean when taking each artist in the collection as seed once.

4.1. Experimental Setup

For our investigation, we opt for an approach comparable to Zobel and Moffat [1998]. A large number of decisions involved in creating artist feature vectors (such as the choice of term frequency $r_{d,t}$ and inverse document frequency w_t), as well as ways to calculate similarity between such feature vectors are evaluated. Most ways to compute these parts originate from previous work in text information retrieval.

4.1.1. Document Modeling/Aggregating Documents. The most central step is the modeling of fundamental text information retrieval concepts such as documents and term frequencies. Once this step is accomplished, known methods to calculate tf (and idf) can be evaluated. In common IR tasks, each document is considered a separate entity. In contrast, in our task each artist is an entity which is represented by a number of documents (i.e., Web pages). There are several ways how to deal with this situation. We evaluate five of them.

(1) *Sum.* All term frequencies appearing in the Web pages associated with the artist are summed up. This corresponds to a simple concatenation of all Web pages related to the artist to one large document.
(2) *Mean.* The term frequency of a term is calculated by taking the arithmetic mean over all pages retrieved for the artist. This is similar to approach 1, but differs in that it is independent of the number of Web pages actually retrieved. Also the range of values is different, which has an impact on some TF calculation approaches.
(3) *Max.* Take the maximum of each term frequency over all retrieved Web pages for the artist.
(4) *NumPagesRel.* Following Whitman [2005], the number of Web pages (retrieved for the artist) that contain the term is used as term frequency. This number is divided by the total number of pages retrieved for the artist.
(5) *NumPagesAbs.* As approach 4, but with the absolute page count, which has an impact on some TF calculation approaches.

We refer to the representation that results from aggregating a number of Web pages retrieved for an artist as *virtual document*.

4.1.2. Page Length Normalization. Based on the idea that Web pages with many terms (i.e., long Web pages) could dominate shorter but nonetheless relevant pages, additionally a normalization step is performed before these aggregation functions are calculated. To minimize interference with the TF calculation approaches (which may depend on the magnitude of the values), the number of terms in each page is normalized to the *page length* (as measured by the sum over the page's raw term frequency count vector). This optional normalization step is done before calculating the TFs, because it intends to simulate pages of same length.

It should be noted that there is another interesting method to combine the Web pages of one artist. It would be possible to calculate the $tf \cdot idf$ value for each Web page separately (i.e, in the initial setup, each Webpage corresponds to one document), and then combine all pages belonging to one artist by a simple aggregation function such as minimum, mean, median or maximum (which may yield different results than mean, subject to the similarity function used). In this case, these functions are calculated *after* having calculated the $tf \cdot idf$ values. We refrain from using this method because the notion our method is based on is to level out page length, a page being either defined as a single Webpage or a virtual artist document (cf. next section). In contrast, that alternative way to combine pages could rather be seen as an attempt to level out different relevances of the retrieved pages. Differing Webpage relevance is not

Table II. Evaluated Variants to Calculate the Term Frequency $r_{d,t}$

Abbr.	Description	Formulation
TF_A	Formulation used for binary match SB = b	$r_{d,t} = \begin{cases} 1 & \text{if } t \in \mathcal{T}_d \\ 0 & \text{otherwise} \end{cases}$
TF_B	Standard formulation SB = t	$r_{d,t} = f_{d,t}$
TF_C	Logarithmic formulation	$r_{d,t} = 1 + \log_e f_{d,t}$
TF_C2	Alternative logarithmic formulation suited for $f_{d,t} < 1$	$r_{d,t} = \log_e(1 + f_{d,t})$
TF_C3	Alternative logarithmic formulation as used in *ltc* variant	$r_{d,t} = 1 + \log_2 f_{d,t}$
TF_D	Normalized formulation	$r_{d,t} = \frac{f_{d,t}}{f_d^m}$
TF_E	Alternative normalized formulation. Similar to Zobel and Moffat [1998] we use $K = 0.5$. SB = n	$r_{d,t} = K + (1 - K) \cdot \frac{f_{d,t}}{f_d^m}$
TF_F	Okapi formulation, according to Zobel and Moffat [1998] and Robertson et al. [1995]. For W we use the vector space formulation, that is, the Euclidean length.	$r_{d,t} = \frac{f_{d,t}}{f_{d,t} + W_d / av_{d \in D}(W_d)}$
TF_G	Okapi BM25 formulation, according to Robertson et al. [1999].	$r_{d,t} = \frac{(k_1+1) \cdot f_{d,t}}{f_{d,t} + k_1 \cdot \left[(1-b) + b \cdot \frac{W_d}{av_{d \in D}(W_d)}\right]}$ $k_1 = 1.2, b = 0.75$

considered in our evaluations, as the retrieval of relevant pages is delegated to the search engine.

4.1.3. Modeling Document Frequency. In the experiments, we opted to model document frequency f_t in two ways. The first way is to regard each virtual artist document as an atomic entity (i.e., N is the number of artists, and f_t is based on the "virtual documents", *vd*). The second way is to take the number of Web pages as the number N of documents and perform the calculation of f_t on individual Web pages (*wp*).

4.1.4. Calculating and Combining tf and idf Weights. In our experiments, nine different methods for calculating the term frequency $r_{d,t}$ are evaluated, as given in Table II. Correspondingly, Table III gives the evaluated methods to calculate the inverse document frequency w_t. Table IV lists the evaluated similarity functions. Disregarding redundant settings,[6] a total of 9,248 different combinations can be defined (by varying the page aggregation function, page length normalization, TF approach, way to model document frequency, IDF approach, and similarity measure). It should be kept in mind that it is likely that the considered functions interfere with the (generally unknown) ranking algorithm used by the search engine, and probably also with the query terms [Knees et al. 2008].

4.1.5. Algorithm Notation. One overall artist similarity algorithm is created by choosing from the options discussed above. In the remainder of this article, we denote such an algorithm in the following way:

```
<PageAggregationFunction>.<PageLengthNormalization>.<TF-Approach>.
      <IDF-Document-Type>.<IDF-Approach>.<SimilarityMeasure>
```

An example of an algorithm in this notation is *Sum.NoPlNorm.TF_A.VirtualDoc.IDF_B2.CosSim*. In cases where a particular choice of variant is clear from the

[6]Note that in some cases, distinct combinations yield the same $tf \cdot idf$ vectors. For example, the value of TF_A is not affected by normalization of pages.

Table III. Evaluated Variants to Calculate the Inverse Document Frequency w_t

Abbr.	Description	Formulation
IDF_A	Formulation used for binary match SB = x	$w_t = 1$
IDF_B	Logarithmic formulation SB = f	$w_t = \log_e\left(1 + \frac{N}{f_t}\right)$
IDF_B2	Logarithmic formulation used in *ltc* variant	$w_t = \log_e\left(\frac{N}{f_t}\right)$
IDF_C	Hyperbolic formulation	$w_t = \frac{1}{f_t}$
IDF_D	Normalized formulation	$w_t = \log_e\left(1 + \frac{f_m}{f_t}\right)$
IDF_E	Another normalized formulation SB = p	$w_t = \log_e \frac{N - f_t}{f_t}$
	The following definitions are based on the term's noise n_t and signal s_t.	$n_t = \sum_{d \in \mathcal{D}_t}\left(-\frac{f_{d,t}}{F_t}\log_2\frac{f_{d,t}}{F_t}\right)$ $s_t = \log_2(F_t - n_t)$
IDF_F	Signal	$w_t = s_t$
IDF_G	Signal-to-Noise ratio	$w_t = \frac{s_t}{n_t}$
IDF_H		$w_t = \left(\max_{t' \in \mathcal{T}} n_{t'}\right) - n_t$
IDF_I	Entropy measure	$w_t = 1 - \frac{n_t}{\log_2 N}$
IDF_J	Okapi BM25 IDF formulation, according to [Robertson et al. 1999; Pérez-Iglesias et al. 2009]	$w_t = \log\frac{N - f_t + 0.5}{f_t + 0.5}$

Table IV. Evaluated Similarity Functions S_{d_1,d_2}

Abbr.	Description	Formulation
INNER	Inner product	$S_{d_1,d_2} = \sum_{t \in \mathcal{T}_{d_1,d_2}}\left(w_{d_1,t} \cdot w_{d_2,t}\right)$
COSSIM	Cosine Measure	$S_{d_1,d_2} = \frac{\sum_{t \in \mathcal{T}_{d_1,d_2}}\left(w_{d_1,t} \cdot w_{d_2,t}\right)}{W_{d_1} \cdot W_{d_2}}$
INNER_ALT	Alternative Inner Product	$S_{d_1,d_2} = \sum_{t \in \mathcal{T}_{d_1,d_2}} \frac{w_{d_2,t}}{W_d}$
DICE	Dice Formulation	$S_{d_1,d_2} = \frac{2\sum_{t \in \mathcal{T}_{d_1,d_2}}\left(w_{d_1,t} \cdot w_{d_2,t}\right)}{W_{d_1}^2 + W_{d_2}^2}$
JACC	Jaccard Formulation	$S_{d_1,d_2} = \frac{\sum_{t \in \mathcal{T}_{d_1,d_2}}\left(w_{d_1,t} \cdot w_{d_2,t}\right)}{W_{d_1}^2 + W_{d_2}^2 - \sum_{t \in \mathcal{T}_{d_1,d_2}}\left(w_{d_1,t} \cdot w_{d_2,t}\right)}$
OVER	Overlap Formulation	$S_{d_1,d_2} = \frac{\sum_{t \in \mathcal{T}_{d_1,d_2}}\left(w_{d_1,t} \cdot w_{d_2,t}\right)}{\min(W_{d_1}^2, W_{d_2}^2)}$
EUCL	Euclidean Similarity	$D_{d_1,d_2} = \sqrt{\sum_{t \in \mathcal{T}_{d_1,d_2}}\left(w_{d_1,t} - w_{d_2,t}\right)^2}$ $S_{d_1,d_2} = \left(\max_{d_1',d_2'}(D_{d_1',d_2'})\right) - D_{d_1,d_2}$
JEFF	Jeffrey Divergence-based Similarity	$S_{d_1,d_2} = \left(\max_{d_1',d_2'}(D_{d_1',d_2'})\right) - D_{d_1,d_2}$ $D(F,G) = \sum_i\left(f_i \log\frac{f_i}{m_i} + g_i \log\frac{g_i}{m_i}\right)$ $m_i = \frac{f_i + g_i}{2}$

context (e.g., when only considering algorithms without page length normalization), the respective part is left out in the notation for brevity.

4.1.6. Term Dictionary. In the literature, there exist a variety of ways to define the terms associated with each dimension of the vector space. To not further complicate the experiments, we opt for using a manually defined dictionary containing 1,379 music-related terms. Assuming that the way of choosing the dictionary avoids common stopwords and terms that appear very infrequently, no downweighting of very frequent and very rare terms is performed. The dictionary comprises terms related to the music domain, such as genre and style descriptors, instruments, epochs, regions, and moods. It was compiled by extracting and merging lists from various Web sources, such as Yahoo! Directory,[7] Wikipedia,[8] and allmusic.com.[9] The list is available for download.[10]

4.1.7. Models Closest to Previous Work. To give a rough orientation how the evaluated techniques are associated with previously used combinations, the closest models to [Whitman and Lawrence 2002; Baumann and Hummel 2003, Knees et al. 2004, Whitman 2005] are given here:

The model closest to Baumann and Hummel [2003] is *Sum.TF_B.NoPlNorm.IDF_B2.CosSim*,[11] and the closest to Knees et al. [2004] is *Sum.TF_C.NoPlNorm.WebPages.IDF_B2.CosSim*, which only uses a different logarithm base. Approach *TF_B.VirtualDoc.IDF_C.Inner* is closest to Whitman and Lawrence [2002] and Whitman [2005]. However, Whitman et al.'s approach seems not easily describable within our framework.

4.2. Evaluation Experiments

Experiments are performed on two sets of artists. The first set (C323a) consists of 323 names of artists from 18 genres drawn from allmusic.com that are assumed to be among the best-known artists in their respective genre. From each genre, approximately the same number of artists was manually selected.

The second set (C3000a), which is more than nine times as large as the first set, comprises 3,000 artist names selected from the music information systems last.fm. We used last.fm's Web API to gather the most popular artists for each country of the world, which we then aggregated into a single list of 201,135 artist names. Since last.fm's data is prone to misspellings or other mistakes due to its collaborative, user-generated knowledge base, we cleaned the data set by matching each artist name with the database of the expert-based music information system allmusic.com, from which we also extracted genre information. Starting this matching process from the most popular artist found by last.fm and including only artist names that also occur in allmusic.com, we retrieved in total 3,000 artists. This number of artists represents the typical size of a current private music collection. Both artist sets are publicly available.[12]

Please note that artist-related Web pages, which constitute the corpus, were determined using the approach presented in the last two paragraphs of Section 3.1.

It is assumed that the best performing $tf \cdot idf$ approaches will do well on both sets. This results in two stages of experiments. In the first stage, all variants are evaluated

[7]http://dir.yahoo.com/Entertainment/Music/Genres.
[8]http://www.wikipedia.org.
[9]http://www.allmusic.com.
[10]http://www.cp.jku.at/people/schedl/music/index_terms_1379.txt.
[11]The paper does not state clearly whether IDF calculation is performed on virtual documents or on individual Web pages.
[12]The first one can be downloaded from http://www.cp.jku.at/people/schedl/music/C323a.txt, the second one is available at http://www.cp.jku.at/people/schedl/music/C3000a.txt.

on the first set. Only the algorithm variants found to perform best in these experiments on the 323 artist set are then evaluated on the larger set in the second stage.

Both sets of artists are divided into the same genre categories, but have different class distributions (the number of artists of the two sets in each genre is given in parentheses): avant garde (19/8), blues (20/11), celtic (12/5), classical (17/42), country (15/24), easy listening (18/6), electronica (18/149), folk (19/24), gospel (18/23), jazz (19/106), latin (15/91), new age (17/18), rnb (20/101), rap (20/203), reggae (20/29), rock (20/2031), vocal (19/30), world (17/99).

Not wanting to go too much into detail at this point, the best-performing combination on the 323-artist-collection was *numPagesAbs.TF_C3.VirtualDoc.IDF_H.CosSim*, the combination that ranked highest on the 3,000-artist-set was *mean.TF_F. VirtualDoc.IDF_B2.Jeff*.

4.2.1. First Stage: Evaluation on the 323-Artist-Set. We model the experiments as a retrieval task. In some major aspects, we follow Buckley and Voorhees [2000] and Sanderson and Zobel [2005]. Given a query artist, the task is to find artists of the same genre via similarity. We use Mean Average Precision (MAP) as the basic performance measure. Average precision is defined as "the mean of the precision scores obtained after each relevant document is retrieved, using zero as the precision for relevant documents that are not retrieved." [Buckley and Voorhees 2000]. Following Sanderson and Zobel [2005], we first calculate MAP of each distinct algorithm variant. These are 9,248 variants. Variants that fulfill both of the following two conditions are discarded.

(1) There is a relative MAP difference of 10% or more to the top-ranked variant,
(2) and the t-test shows a significant difference to the top-ranked variant.

When doing so, and subsequently ranking all 9,248 variants according to MAP, the top 123 variants have a relative MAP difference (from the 1^{st} to the respective rank) of less than 10%. A pairwise t-test shows a significant difference for all variants except for the topmost 134 variants and the 136^{th} ranked variant. This sharp cutoff of nonsignificant vs. significant results and the relatively high accordance of our two criteria (less than 10% MAP difference and significance) supports our reasoning that these top-ranked algorithms are those worth further examination. A detailed list of the MAP scores for the best- and worst-performing variants is given in Table V.

As for the *BM25* weighting, that is, the combination of TF_G (cf. Table II) and IDF_J (cf. Table III), variant *numpagesrel.none.TF_G.wp.IDF_J.COSSIM* as best-performing combination is ranked at position 141, therefore slightly below the threshold for the MAP difference of 10% to the top-ranked variant. Although this variant is hence not included for the second stage of experiments, it is noteworthy that the *BM25* measure works best when calculating the IDF values on the level of individual Web pages, instead of modeling virtual documents.

To get more insight into which components are of high value, we look at each of the algorithm's components separately, and examine which approaches appear in the 135 selected algorithms, and how often they appear. First, it becomes apparent that only variants based on *un*normalized Web page lengths appear in the top-ranked variants. Thus, normalization does not seem to improve performance. Also, only idf computation approaches based on virtual documents are encountered. Therefore, calculating the inverse document frequency on Web pages instead of artist level in general seems not beneficial.

Figures 1 to 4 show histograms of the remaining algorithm components (page aggregation function, TF method, IDF method, and similarity measure). Note that weak performing variants have been omitted, as already described. The figures give a first insight into the relative performance of the different variants. The algorithm

Table V. MAP Scores of the Top-Ranked Variants (Notation as Described in Section 4.1.5).

MAP	Variant
0.38732	numpagesabs.none.TF_C3.vd.IDF_H.COSSIM
0.38642	numpagesabs.none.TF_C3.vd.IDF_I.COSSIM
0.38624	numpagesabs.none.TF_C2.vd.IDF_H.COSSIM
0.38523	numpagesabs.none.TF_C2.vd.IDF_I.COSSIM
0.37855	numpagesrel.none.TF_F.vd.IDF_H.COSSIM
0.37854	numpagesrel.none.TF_F.vd.IDF_I.COSSIM
0.37788	numpagesabs.none.TF_C.vd.IDF_H.COSSIM
0.37780	numpagesabs.none.TF_C.vd.IDF_I.COSSIM
0.37728	numpagesrel.none.TF_F.vd.IDF_B2.COSSIM
0.37692	mean.none.TF_F.vd.IDF_E.JEFF
0.37446	mean.none.TF_C2.vd.IDF_E.JEFF
0.37302	sum.none.TF_C2.vd.IDF_B2.COSSIM
0.37299	sum.none.TF_C2.vd.IDF_B2.JACC
0.37299	sum.none.TF_C2.vd.IDF_B2.DICE
0.37076	sum.none.TF_C3.vd.IDF_B2.COSSIM
0.37059	sum.none.TF_C3.vd.IDF_B2.JACC
0.37059	sum.none.TF_C3.vd.IDF_B2.DICE
0.37050	mean.none.TF_F.vd.IDF_B2.JEFF
0.36918	numpagesrel.none.TF_C2.vd.IDF_B2.COSSIM
0.36896	numpagesrel.none.TF_C2.vd.IDF_H.COSSIM
0.36895	numpagesrel.none.TF_C2.vd.IDF_I.COSSIM
0.36806	numpagesrel.none.TF_F.vd.IDF_I.JACC
0.36806	numpagesrel.none.TF_F.vd.IDF_I.DICE
0.36805	numpagesrel.none.TF_F.vd.IDF_H.JACC
0.36805	numpagesrel.none.TF_F.vd.IDF_H.DICE
0.36758	numpagesabs.none.TF_C2.vd.IDF_H.JACC
0.36758	numpagesabs.none.TF_C2.vd.IDF_H.DICE
0.36685	numpagesabs.none.TF_C2.vd.IDF_I.JACC
0.36685	numpagesabs.none.TF_C2.vd.IDF_I.DICE
0.36629	sum.none.TF_C2.vd.IDF_I.COSSIM
...	...
0.01097	mean.none.TF_B.vd.IDF_F.OVER
0.01097	mean.none.TF_D.vd.IDF_F.OVER
0.01081	mean.none.TF_B.vd.IDF_C.OVER
0.01075	mean.none.TF_B.vd.IDF_B.OVER
0.01075	mean.none.TF_B.vd.IDF_D.OVER
0.01055	mean.none.TF_B.vd.IDF_G.OVER
0.01044	mean.sum.TF_B.wp.IDF_F.OVER
0.01015	mean.sum.TF_B.vd.IDF_F.OVER
0.00952	mean.sum.TF_B.wp.IDF_A.OVER
0.00952	mean.sum.TF_B.vd.IDF_A.OVER

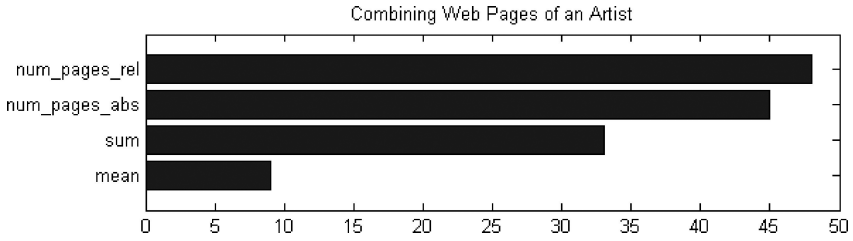


Fig. 1. Methods to combine terms appearing on an artist's Web pages. Only those appearing in the 135 selected top algorithms are shown, and the number of times they appear (totaling 135).

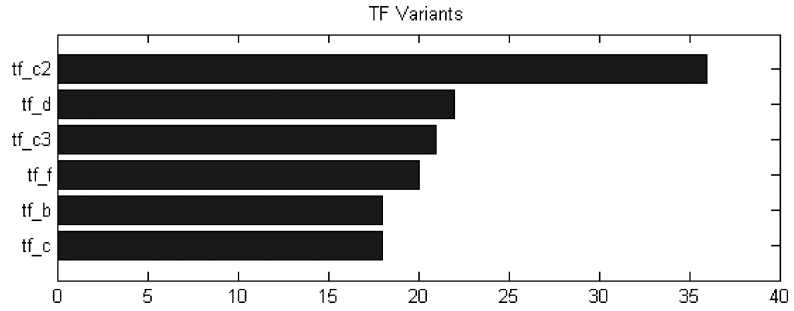


Fig. 2. TF approaches appearing in the 135 selected top algorithms, and the number of times they appear (totaling 135).

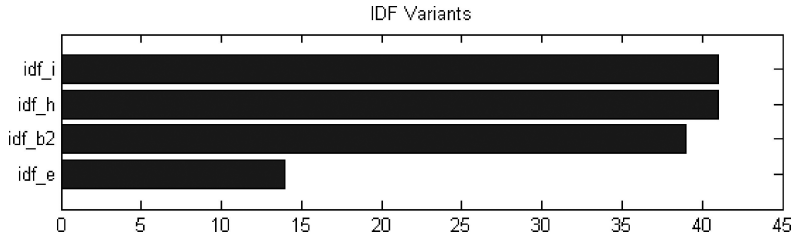


Fig. 3. IDF variants appearing in the 135 selected top algorithms, and the number of times they appear (totaling 135).

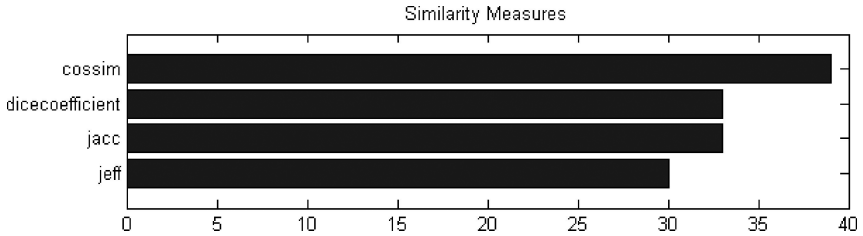


Fig. 4. Similarity measures appearing in the 135 selected top algorithms, and the number of times they appear (totaling 135).

representing the most frequently appearing variant of each component (i.e., numPages-Rel – TF_C2 – IDF_I – CosineSimilarity) is interestingly ranked only 21^{th} in the overall ranking.

However, it cannot be assumed that the shown frequencies are mutually independent. For example, when for one component of the algorithm two highly similar variants are evaluated, the other components that perform well in combination with these variants will appear more frequently.

Hence, instead of analyzing the figures more deeply, we go on by evaluating on the second set consisting of 3,000 artists all possible algorithms that can be created with the remaining variants. Thus, the only assumption is that variants that do not appear in the set of the 135 selected algorithms are not well suited for our desired algorithm to compute artist similarity. In detail, additionally to normalizing Web page length and calculating document frequency on the Web page level, the variants that are discarded here are as follows.

—*Document modeling. max*, that is, taking the maximum number of appearances over all Web pages of an artist.
—*tf computation.* variant *A* (*binary match*, i.e., if a term is contained in a document or not) and variant *E* ("alternative normalized formulation").
—*idf computation.* variants *A, B, C, D, F, G* (cf. Table III).
—*Similarity measure.* variant *INNER* (inner product), *INNER_ALT* (alternative inner product), *OVERLAP* (overlap formulation), *EUCL* (Euclidean similarity).

The remaining variants can be used to create 384 different combinations of $tf \cdot idf$ approaches and similarity measures.

4.2.2. Second Stage: Evaluation on the 3,000-Artist-Set. Since we further aim at evaluating the various approaches on a real-world collection, we retrieved the most popular artists as of the end of February 2010 from last.fm, as previously described.

In the second stage of the evaluation experiments, this 3,000-artist-set is used to investigate if both artist sets yield a comparable ranking of the 384 algorithms of interest, and which of these algorithms are top-ranked on both sets of artists. To clarify the first aspect, Spearman's rank-order correlation coefficient [Sheskin 2004] is computed on the two rankings obtained with the two artist sets. This experiment shows a correlation coefficient of 0.91. This high correlation indicates that, in general, the ranking of the algorithms is not largely influenced by factors such as size of artist collection and number of artists per genre. We note, however, that both artist sets contain mainly popular artists.

To get insight into which out of the 384 algorithms are top-ranked on both sets of artists, a ranked list of the best performing algorithms is created. In this list, algorithms are sorted based on their maximum (i.e., lowest numeric) rank in either of the two experiments (the two artist sets). For example, if an algorithm ranked second in the algorithm ranking based on the set of 323 artists, and 15^{th} on the set of 3,000 artists, then the value associated with this algorithm is 15. The corresponding list is given in the appendix.

As can be seen from the list, the $tf \cdot idf$ algorithm used in Baumann and Hummel [2003], applied to our data sets, has a maximum rank of 319. The algorithm from Knees et al. [2004] does not appear in the list, as it uses the number of Web pages to determine the document frequency, which was outside the significance bounds in the first stage. However, approach *sum.TF_C3.IDF_B2.CosSim*, which has a maximum rank of 17 in the two experiments, resembles this algorithm (counting the number of artists instead of counting the number of Web pages a term appears on). This may be seen as an indication that this variant is a good choice for the considered area of application, and it also shows that changing only one factor can have an important impact on the performance of an algorithm. Based on the latter observation, it seems that no valid statement about the relative performance of the algorithms used in Whitman and Lawrence [2002] and Whitman [2005] can be made, as the exact similarity measure used there was not evaluated in our experiments.

To gain better insights into the distribution of the different variants for the decisions regarding the algorithmic components, we show the occurrences of each algorithm variant among the various ranks (from 1 to 384). Instead of showing binary values (i.e., black for occurring/white for not occurring), for (assumed) better visibility we smoothed values by kernel density estimation. The results are reported in Figures 5 (for different aggregation functions), Figure 6 (for different term frequency formulations), Figure 7 (for different inverse document frequency), and Figure 8 (for different similarity measures). The figures' x-axes depict at which ranks the respective variants occur. Darker values indicate that the respective variant occurs more frequently in the corresponding range of ranks, while bright values indicate that the respective variant does less

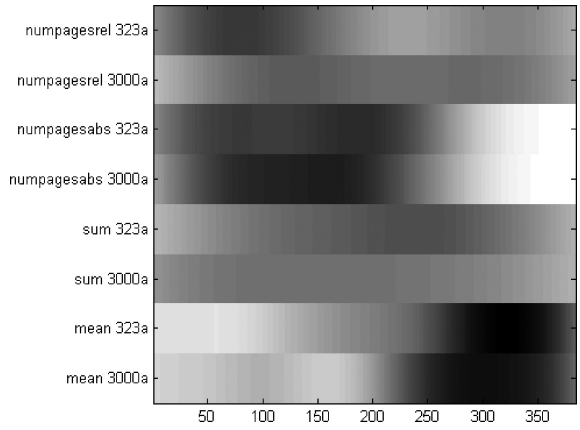


Fig. 5. Kernel density estimation for different aggregation functions.

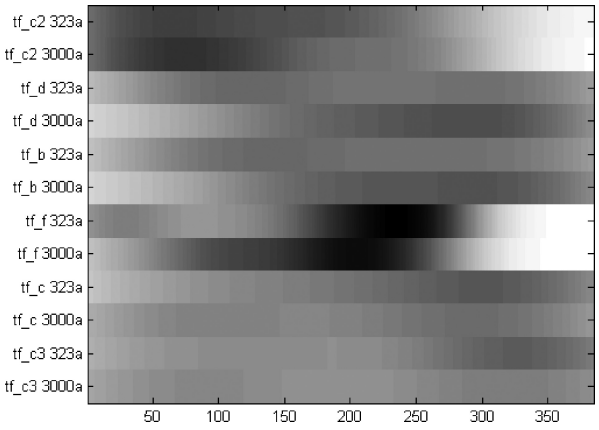


Fig. 6. Kernel density estimation for different term frequency formulations.

frequently occur in this range of ranks. From the figures, we can see a certain tendency of wide spreads in the distribution of individual variants. For example, considering Figure 7 reveals that the two best performing *idf* variants (`H` and `I`) occur in a wide range of ranks. Figure 5 demonstrates that using the mean as aggregation function is a comparably bad choice (relative to the other selected variants). From Figure 6 we can see that variant `C2` for the *tf* calculation outperforms the others considerably. Looking at Figure 7 gives no clear picture as the best performing *idf* variants `H` and `I` occur among a widespread range of ranks. As for the different similarity measures (Figure 8), although the Dice and the Jaccard coefficient performed best on average, upper ranks on C323a are dominated by the cosine measure, whereas on C3000a the Jeffrey divergence appears most frequently. In general, we can see that the cosine measure yields the most stable results, which means that the overall performance of a music similarity measure is least influenced when using the cosine measure.

5. CONCLUSIONS AND FUTURE WORK

Relative to our evaluation setting, the conclusions for calculating artist similarity can be summarized as follows. A minor finding is that normalization of each Web page

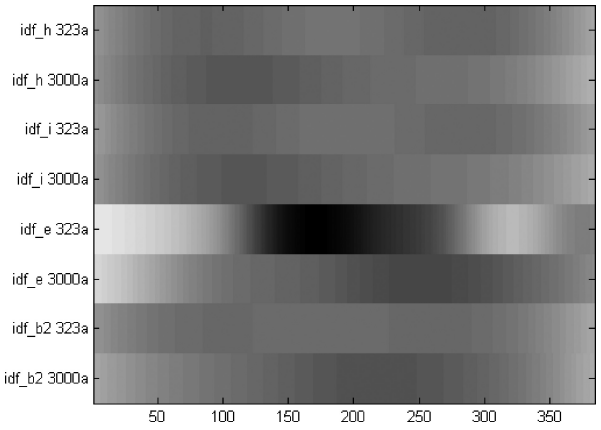


Fig. 7. Kernel density estimation for different inverse document frequency formulations.

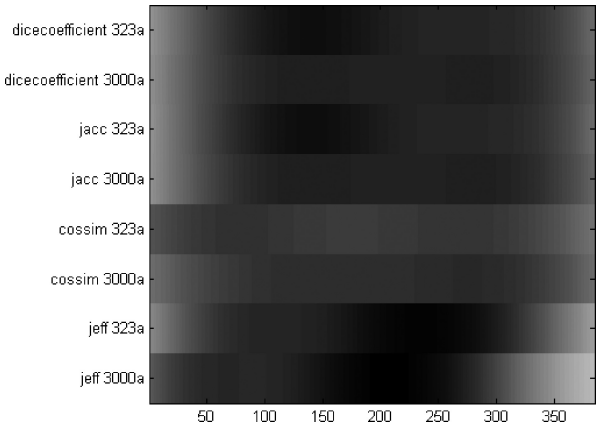


Fig. 8. Kernel density estimation for different similarity measures.

(so that each Web page has the same total weight) showed not to be of benefit. It seems, however, much more important that the document frequency for calculating idf is determined on virtual documents rather than on individual Web pages. Additionally, a number of possible variants did not appear in the top-ranked algorithms in the first stage of our experiments, conducted on the C323a set. Assuming that the best results are obtained when using the remaining variants, it is possible to prune the space of possible algorithms from 9,248 to 384 candidate algorithms. The frequently used cosine similarity measure appears for many of these top-ranked algorithms. However, while it was the measure in the highest ranked algorithm on the 323 artists set (*numPagesAbs.TF_C3.VirtualDoc.IDF_H.CosSim*), the algorithm that ranked highest on the 3,000-artist-set was *mean.TF_F.VirtualDoc.IDF_B2.Jeff*. Factors concerning the collection, such as size of the collection and number of artists per genre, seem to have only a minor impact on the relative performance of the best algorithms, as far as can be concluded from the evaluated parameter ranges. In contrast, a small change to an algorithm (document frequency calculated on Web pages vs. on artist level) can have an important impact on the algorithm's relative performance. The latter observation

encourages further evaluation of different text processing approaches, different term sets for indexing, term selection and term weighting functions.

On the other hand, in accordance with Zobel and Moffat [1998], we have to admit that we were not able to distill a specific combination out of the remaining 384 algorithms that worked best for both test collections, neither can we report on a choice for individual aspects (e.g., variant of term frequency, variant of similarity measure) that always outperformed all other variants. The interdependencies between different decisions which variants to choose for the individual components seem to be too large to obtain an overall winning combination. Thus, Zobel and Moffat's final statement, "The measures do not form a space that can be explored in any meaningful way, other than by exhaustion," does unfortunately also apply analogously to the music similarity space derived from music-related Web pages. But considering that we are able to restrict this space to 384 candidate algorithms in our evaluation setting, exhaustion within this subspace seems feasible.

This study focused on the task of (text-based) similarity estimation between music artists, which is a relatively specific, nevertheless important, task in music information research. Other MIR tasks such as artist clustering, text-based music retrieval, or automated playlist generation might require other formulations of algorithm variants. It seems reasonable to conclude that, depending on the task, various parameter choices need to be evaluated. Nevertheless, the results of this study may support research towards personalized music retrieval as well as combining different aspects of music similarity. For example, Zhang et al. [2009] propose a system for personalized music search, taking into account similarity aspects derived from music content and from social factors. A multimodal music similarity model taking subjective aspects into account is also presented in McFee and Lanckriet [2009]. Since such work on personalized MIR systems is strongly related to text-based representation of (music-related) documents—not only of artist pages, but also of user-generated content (e.g., instant messages or social network posts)—efficient term weighting and similarity measures are crucial. Furthermore, approaches that combine content-based with context-based information for the purpose of music playlist generation, such as Pohle et al. [2007], are likely to benefit from the results of this study.

As for future work, the current experiments are limited to the rather narrow task of genre classification. The genre assignment of the two test collections used originates from allmusic.com's experts' judgments. A possibly more accurate ground truth could be derived from "similar artist"-relations given by last.fm's collaborative filtering approach. Even though this data is likely prone to a population bias and information may be sparse [Schedl and Knees 2009], evaluation against such a ground truth definition may yield interesting findings.

Another direction in which to extend the work at hand is determining the influence of individual choices made in the analyzed variants for normalization, aggregation, *tf* and *idf* formulations, and similarity measurement. To this end, a general linear regression model could be used to assess the relative impact of various decisions.

APPENDIX : DETAILED RESULTS

In the following, a sorted list of the best performing approaches is given. The number gives the lower of the two ranks (obtained on the 323-artist-set and the 3,000-artist-set). The list contains all combinations that differed not significantly from the respective best variant – neither on the 323-artist-set, nor on the 3,000-artist-set. Entries have the form `<PageAggregationFunction>.<TF-Approach>.<IDF-Approach>.<SimilarityMeasure>`.

6. numpagesabs.tf_c3.idf_i.cossim
8. numpagesabs.tf_c3.idf_h.cossim
9. numpagesabs.tf_c2.idf_i.cossim
10. numpagesabs.tf_c2.idf_h.cossim
13. mean.tf_f.idf_e.jeff
15. sum.tf_c2.idf_b2.cossim
16. mean.tf_f.idf_b2.jeff
17. sum.tf_c3.idf_b2.cossim
18. sum.tf_c2.idf_b2.dice
19. sum.tf_c2.idf_b2.jacc
22. numpagesabs.tf_c.idf_i.cossim
23. numpagesabs.tf_c.idf_h.cossim
31. sum.tf_c2.idf_i.cossim
32. sum.tf_c2.idf_h.cossim
35. sum.tf_c3.idf_b2.dice
35. sum.tf_c3.idf_i.cossim
36. sum.tf_c3.idf_b2.jacc
36. numpagesrel.tf_c2.idf_b2.jeff
37. numpagesabs.tf_b.idf_b2.jeff
38. numpagesabs.tf_d.idf_b2.jeff
39. numpagesrel.tf_b.idf_b2.jeff
40. numpagesrel.tf_d.idf_b2.jeff
41. mean.tf_c2.idf_b2.jeff
44. sum.tf_c3.idf_h.cossim
47. numpagesrel.tf_f.idf_b2.jeff
48. sum.tf_c.idf_b2.cossim
51. sum.tf_c.idf_i.cossim
55. sum.tf_c.idf_b2.dice
56. sum.tf_c.idf_b2.jacc
57. numpagesabs.tf_c2.idf_h.dice
58. numpagesabs.tf_c2.idf_h.jacc
59. numpagesabs.tf_c3.idf_b2.cossim
60. numpagesabs.tf_c2.idf_i.dice
61. numpagesabs.tf_c2.idf_i.jacc
62. sum.tf_c.idf_h.cossim
62. numpagesabs.tf_c3.idf_h.dice
63. numpagesabs.tf_c3.idf_h.jacc
64. numpagesabs.tf_c3.idf_i.dice
65. numpagesabs.tf_c3.idf_i.jacc
68. numpagesrel.tf_f.idf_i.cossim
69. numpagesrel.tf_f.idf_h.cossim
70. numpagesabs.tf_c.idf_h.dice
71. numpagesabs.tf_c.idf_h.jacc
71. numpagesabs.tf_c2.idf_b2.cossim
74. numpagesabs.tf_c.idf_i.dice
75. numpagesabs.tf_c.idf_i.jacc
75. numpagesrel.tf_b.idf_i.jeff
76. numpagesrel.tf_d.idf_i.jeff
77. numpagesrel.tf_b.idf_h.jeff
78. numpagesabs.tf_c3.idf_b2.dice
78. numpagesrel.tf_c2.idf_i.jeff
79. numpagesabs.tf_c3.idf_b2.jacc
79. numpagesrel.tf_d.idf_h.jeff
80. numpagesrel.tf_c2.idf_h.jeff
81. numpagesabs.tf_b.idf_h.jeff
82. mean.tf_c2.idf_e.jeff
82. numpagesabs.tf_d.idf_h.jeff
85. numpagesrel.tf_f.idf_h.jeff
86. numpagesrel.tf_c2.idf_e.jeff
86. numpagesrel.tf_f.idf_i.jeff
87. numpagesrel.tf_f.idf_e.jeff
88. numpagesabs.tf_b.idf_i.jeff
89. numpagesabs.tf_b.idf_e.jeff
89. numpagesabs.tf_d.idf_i.jeff
90. mean.tf_f.idf_h.jeff
90. numpagesabs.tf_d.idf_e.jeff
91. sum.tf_c2.idf_h.dice
91. numpagesrel.tf_b.idf_e.jeff
92. sum.tf_c2.idf_h.jacc
92. numpagesrel.tf_d.idf_e.jeff
93. sum.tf_c2.idf_i.dice
93. numpagesrel.tf_f.idf_h.dice
94. sum.tf_c2.idf_i.jacc
94. numpagesrel.tf_f.idf_h.jacc
95. numpagesabs.tf_c2.idf_b2.dice
95. numpagesrel.tf_f.idf_i.dice
96. numpagesabs.tf_c2.idf_b2.jacc
96. numpagesrel.tf_f.idf_i.jacc
97. numpagesrel.tf_c2.idf_h.cossim
98. sum.tf_c3.idf_h.dice
98. numpagesrel.tf_c2.idf_i.cossim
99. sum.tf_c3.idf_h.jacc
99. numpagesrel.tf_f.idf_b2.cossim
100. sum.tf_c3.idf_i.dice
100. numpagesrel.tf_c2.idf_h.dice
101. sum.tf_c3.idf_i.jacc
101. numpagesrel.tf_c2.idf_h.jacc
102. sum.tf_c.idf_h.dice
102. numpagesrel.tf_c2.idf_i.dice
103. sum.tf_c.idf_h.jacc
103. numpagesrel.tf_c2.idf_i.jacc
104. mean.tf_f.idf_i.jeff
104. numpagesabs.tf_b.idf_h.cossim
105. numpagesabs.tf_c.idf_b2.cossim
105. numpagesabs.tf_d.idf_h.cossim
106. sum.tf_c.idf_i.dice
106. numpagesrel.tf_b.idf_h.cossim
107. sum.tf_c.idf_i.jacc
107. numpagesrel.tf_d.idf_h.cossim
108. sum.tf_c3.idf_e.dice
108. numpagesrel.tf_b.idf_i.cossim
109. sum.tf_c3.idf_e.jacc
109. numpagesrel.tf_d.idf_i.cossim
110. mean.tf_f.idf_b2.cossim
110. sum.tf_c2.idf_e.dice
111. sum.tf_c2.idf_e.jacc
111. numpagesabs.tf_b.idf_i.cossim
112. numpagesabs.tf_c.idf_b2.dice
112. numpagesabs.tf_d.idf_i.cossim
113. numpagesabs.tf_c.idf_b2.jacc
113. numpagesrel.tf_c2.idf_b2.cossim
114. sum.tf_c2.idf_e.cossim
114. numpagesrel.tf_f.idf_b2.dice
115. sum.tf_c3.idf_e.cossim
115. numpagesrel.tf_f.idf_b2.jacc

All variants do not use page length normalization, and the document frequency is always calculated on virtual documents (and not taken as the number of Web pages). For brevity, these choices are not mentioned explicitly.

REFERENCES

AHN, L. V. AND DABBISH, L. 2004. Labeling images with a computer game. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*.

AUCOUTURIER, J.-J. AND PACHET, F. 2002. Scaling up music playlist generation. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'02)*. 105–108.

AUCOUTURIER, J.-J. AND PACHET, F. 2004. Improving timbre similarity: How high is the sky? *J. Neg. Results Speech Audio Sci. 1,* 1.

BACCIGALUPO, C., PLAZA, E., AND DONALDSON, J. 2008. Uncovering affinity of artists to multiple genres from social behaviour data. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR'08)*.

BAEZA-YATES, R. AND RIBEIRO-NETO, B. 1999. *Modern Information Retrieval*. Addison Wesley.

BAUMANN, S. AND HUMMEL, O. 2003. Using cultural metadata for artist recommendation. In *Proceedings of the Conference on Web Delivering of Music (WEDELMUSIC'02)*.

BERENZWEIG, A., LOGAN, B., ELLIS, D. P., AND WHITMAN, B. 2003. A large-scale evaluation of acoustic and subjective music similarity measures. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR'03)*.

BUCKLEY, C. AND VOORHEES, E. 2000. Evaluating evaluation measure stability. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

CASEY, M. A., VELTKAMP, R., GOTO, M., LEMAN, M., RHODES, C., AND SLANEY, M. 2008. Content-based music information retrieval: Current directions and future challenges. *Proc. IEEE 96*, 668–696.

CELMA, O., CANO, P., AND HERRERA, P. 2006. Search sounds: An audio crawler focused on weblogs. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR'06)*.

CELMA, O. AND LAMERE, P. 2007. ISMIR 2007 Tutorial: Music recommendation. http://mtg.upf.edu/~ocelma/MusicRecommendationTutorial-ISMIR2007 (last accessed 12/07).

CHAKRABARTI, S., VAN DEN BERG, M., AND DOM, B. 1999. Focused crawling: A new approach to topic-specific web resource discovery. *Comput. Netw. 31,* 11–16, 1623–1640.

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'04)*. ACM Press, New York, NY, 462–471.

CIMIANO, P. AND STAAB, S. 2004. Learning by Googling. *ACM SIGKDD Explor. Newsle. 6,* 2, 24–33.

COHEN, W. W. AND FAN, W. 2000a. Web-collaborative filtering: Recommending music by crawling the Web. *Comput. Netw. 33,* 1–6, 685–698.

DEERWESTER, S., DUMAIS, S. T., FURNAS, G. W., LANDAUER, T. K., AND HARSHMAN, R. 1990. Indexing by latent semantic analysis. *J. Am. Soc. Inform. Sci. 41*, 391–407.

DOWNIE, J. S. 2003. Toward the scientific evaluation of music information retrieval systems. In *Proceedings of the 4th International Symposium on Music Information Retrieval (ISMIR'03)*.

ELLIS, D. P. W. 2002. The quest for ground truth in musical artist similarity. In *Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR'02)*.

FINGERHUT, M. 2004. Music information retrieval, or how to search for (and maybe find) music and do away with incipits. Slides for IAML/IASA Congress.

GELEIJNSE, G. AND KORST, J. 2006. Web-based artist categorization. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR'06)*.

GÖKER, A. AND MYRHAUG, H. I. 2002. User context and personalisation. In *Proceedings of the 6th European Conference on Case Based Reasoning (ECCBR'02)* (Workshop on Case Based Reasoning and Personalization).

GOVAERTS, S. AND DUVAL, E. 2009. A Web-based approach to determine the origin of an artist. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR'09)*.

HOFMANN, T. 1999. Probabilistic latent semantic analysis. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*.

HU, X., DOWNIE, J. S., AND EHMANN, A. F. 2009. Lyric text mining in music mood classification. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR'09)*.

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'05)*.

KASSLER, M. 1966. Musical information retrieval. *Perspect. New Music* 4, 59–67.

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'04)*. 517–524.

KNEES, P., POHLE, T., SCHEDL, M., AND WIDMER, G. 2007. 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'07)*.

KNEES, P., SCHEDL, M., AND POHLE, T. 2008. A deeper look into Web-based classification of music artists. In *Proceedings of the 2nd Workshop on Learning the Semantics of Audio Signals (LSAS'08)*.

KNEES, P., SCHEDL, M., POHLE, T., AND WIDMER, G. 2007. Exploring music collections in virtual landscapes. *IEEE MultiMed. 14,* 3, 46–54.

LAURIER, C., GRIVOLLA, J., AND HERRERA, P. 2008. Multimodal music mood classification using audio and lyrics. In *Proceedings of the International Conference on Machine Learning and Applications*.

LAW, E. L. M., VON AHN, L., DANNENBERG, R. B., AND CRAWFORD, M. 2007. Tagatune: A game for music and sound annotation. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR'07)*.

LOGAN, B., ELLIS, D. P. W., AND BERENZWEIG, A. 2003. Toward evaluation techniques for music similarity. In *Proceedings of the Workshop on the Evaluation of Music Information Retrieval (MIR) Systems at SIGIR*.

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'04)*.

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'05)*. 475–478.

MANDEL, M. I. AND ELLIS, D. P. W. 2007. A web-based game for collecting music metadata. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR'07)*.

MCFEE, B. AND LANCKRIET, G. 2009. Heterogeneous embedding for subjective artist similarity. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR'09)*.

PACHET, F. AND CAZALY, D. 2000. A taxonomy of musical genre. In *Proceedings of Content-Based Multimedia Information Access (RIAO) Conference*.

PACHET, F., WESTERMANN, 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'01)*.

PAMPALK, E. 2006. Computational models of music similarity and their application to music information retrieval. Ph.D. thesis, Vienna University of Technology.

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'05)*.

PAMPALK, E. AND GOTO, M. 2007. MusicSun: A new approach to artist recommendation. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR'07)*.

PAMPALK, E., RAUBER, A., AND MERKL, D. 2002. Content-based organization and visualization of music archives. In *Proceedings of the 10th ACM International Conference on Multimedia (MM'02)*. 570–579.

PÉREZ-IGLESIAS, J., PÉREZ-AGÜERA, J. R., FRESNO, V., AND FEINSTEIN, Y. Z. 2009. Integrating the probabilistic models BM25/BM25F into Lucene. CoRR abs/0911.5046.

POHLE, T. 2009. Automatic characterization of music for intuitive retrieval. Ph.D. thesis, Johannes Kepler University Linz, Austria.

POHLE, T., KNEES, P., SCHEDL, M., PAMPALK, E., AND WIDMER, G. 2007c. "Reinventing the Wheel": A novel approach to music player interfaces. *IEEE Trans. Multimed. 9*, 567–575.

POHLE, T., KNEES, P., SCHEDL, M., AND WIDMER, G. 2007a. 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)*.

POHLE, T., KNEES, P., SCHEDL, M., AND WIDMER, G. 2007b. Meaningfully browsing music services. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR'07)*.

ROBERTSON, S., WALKER, S., AND BEAULIEU, M. 1999. Okapi at TREC-7: Automatic ad hoc, filtering, VLC and interactive track. In *Proceedings of the 7th Text REtrieval Conference*. 253–264.

ROBERTSON, S., WALKER, S., AND HANCOCK-BEAULIEU, M. 1995. Large test collection experiments on an operational, interactive system: Okapi at TREC. *In Inform. Process. Manage. 31*, 345–360.

THORPE, S., FIZE, D. , AND MARLOT, C. 1996. Speed of processing in the human visual system. *Nature 381,* 6582, 520–522.

Sanderson, M. and Zobel, J. 2005. Information retrieval system evaluation: Effort, sensitivity, and reliability. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'05)*.

Schedl, M. 2008. Automatically extracting, analyzing, and visualizing information on music artists from the World Wide Web. Ph.D. thesis, Johannes Kepler University Linz, Austria.

Schedl, M. and Knees, P. 2009. Context-based music similarity estimation. In *Proceedings of the 3rd International Workshop on Learning the Semantics of Audio Signals (LSAS'09)*.

Schedl, M., Pampalk, E., and Widmer, G. 2005. Intelligent structuring and exploration of digital music collections. *e&i—Elektrotechnik und Informationstechnik 122,* 7–8, 232–237.

Schedl, M. and Pohle, T. 2010. Enlightening the sun: A user interface to explore music artists via multimedia content. *Multimed. Tools Appl. 49*, 1, (Special Issue on Semantic and Digital Media Technologies) 101–118.

Schedl, M., Seyerlehner, K., Widmer, G., and Schiketanz, C. 2010. Three Web-based heuristics to determine a person's or institution's country of origin. In *Proceedings of the 33th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'10)*.

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'07)*.

Schedl, M., Widmer, G., Pohle, T., and Seyerlehner, K. 2007. Web-based detection of music band members and line-up. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR'07)*.

Schnitzer, D., Pohle, T., Knees, P., and Widmer, G. 2007. One-touch access to music on mobile devices. In *Proceedings of the 6th International Conference on Mobile and Ubiquitous Multimedia (MUM'07)*.

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)*.

Shavitt, Y. and Weinsberg, U. 2009. Songs clustering using peer-to-peer co-occurrences. In *Proceedings of the IEEE International Symposium on Multimedia (ISM'09): International Workshop on Advances in Music Information Research (AdMIRe'09)*.

Sheskin, D. J. 2004. *Handbook of Parametric and Nonparametric Statistical Procedures*, 3rd Ed. Chapman & Hall/CRC, Boca Raton.

Stenzel, R. and Kamps, T. 2005. Improving content-based similarity measures by training a collaborative model. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR'05)*.

Turnbull, D., Barrington, L., and Lanckriet, G. 2008. Five approaches to collecting tags for music. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR'08)*.

Turnbull, D., Liu, R., Barrington, L., and Lanckriet, G. 2007. A game-based approach for collecting semantic annotations of music. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR'07)*.

Whitman, B. 2005. Learning the meaning of music. Ph.D. thesis, School of Architecture and Planning, Massachusetts Institute of Technology, Cambridge, MA.

Whitman, B. and Lawrence, S. 2002. Inferring descriptions and similarity for music from community metadata. In *Proceedings of the International Computer Music Conference (ICMC)*. 591–598

Zadel, M. and Fujinaga, I. 2004. Web services for music information retrieval. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR'04)*.

Zhang, B., Shen, J., Xiang, Q., and Wang, Y. 2009. CompositeMap: A novel framework for music similarity measure. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'09)*. ACM, New York, NY, 403–410.

Zobel, J. and Moffat, A. 1998. Exploring the similarity space. *ACM SIGIR Forum 32,* 1, 18–34.

Zobel, J. and Moffat, A. 2006. Inverted files for text search engines. *ACM Comput. Surv. 38*, 1–56.

Received May 2010; revised November 2010; accepted January 2011

Markus Schedl

#nowplaying Madonna: A Large-Scale Evaluation on Estimating Similarities Between Music Artists and Between Movies from Microblogs

Information Retrieval, 15:183–217, June 2012

Inf Retrieval
DOI 10.1007/s10791-012-9187-y

INFORMATION RETRIEVAL FOR SOCIAL MEDIA

#nowplaying Madonna: a large-scale evaluation on estimating similarities between music artists and between movies from microblogs

Markus Schedl

Received: 1 April 2011 / Accepted: 23 January 2012

Abstract Different term weighting techniques such as *TF · IDF* or *BM25* have been used intensely for manifold text-based information retrieval tasks. Their use for modeling term profiles for named entities and subsequent calculation of similarities between these named entities have been studied to a much smaller extent. The recent trend of microblogging made available massive amounts of information about almost every topic around the world. Therefore, microblogs represent a valuable source for text-based named entity modeling. In this paper, we present a systematic and comprehensive evaluation of different *term weighting measures*, *normalization techniques*, *query schemes*, *index term sets*, and *similarity functions* for the task of inferring similarities between named entities, based on data extracted from *microblog posts*. We analyze several thousand combinations of choices for the above mentioned dimensions, which influence the similarity calculation process, and we investigate in which way they impact the quality of the similarity estimates. Evaluation is performed using three real-world data sets: two collections of microblogs related to music artists and one related to movies. For the music collections, we present results of *genre classification experiments* using as benchmark genre information from `allmusic.com`. For the movie collection, we present results of *multi-class classification experiments* using as benchmark categories from `IMDb`. We show that microblogs can indeed be exploited to model named entity similarity with remarkable accuracy, provided the correct settings for the analyzed aspects are used. We further compare the results to those obtained when using Web pages as data source.

Keywords Social media mining · Microblog analysis · Vector space model · Term weighting · Information extraction · Evaluation

M. Schedl (✉)
Department of Computational Perception, Johannes Kepler University,
Altenberger Straße 69, 4040 Linz, Austria
e-mail: markus.schedl@jku.at

Published online: 06 March 2012

1 Introduction

Microblogging has encountered a tremendous popularity gain during the past couple of years. Today's most popular microblogging service `Twitter`[1] has more than 100 million registered users (Yarow 2011). Millions of users post "tweets" that reveal what they are doing, what is on their mind, or what is currently important for them. According to Evans (2011), the number of tweets per day surpassed 50 millions in early 2010. `Twitter` thus represents a rich data source for text-based information extraction (IE) and information retrieval (IR).

In classical text-IR, term weighting techniques such as $TF \cdot IDF$ and *BM25* are typically used in combination with a similarity function to estimate the relevance of a set of documents to a query. In IE the same techniques (term weighting and similarity calculation) can be used to model term profiles for named entities and compute pairwise similarity scores between these entities. Such similarity measures are vital for various applications, in particular, in the domain of multimedia retrieval. For example, in music information retrieval elaborating musical similarity measures that are capable of capturing aspects that relate to real, perceived similarity is one of the main challenges as it enables a wealth of intelligent music applications. Examples are systems to automatically generate playlists (Aucouturier and Pachet 2002; Pohle et al. 2007), music recommender systems (Celma 2008; Zadel and Fujinaga 2004), music information systems (Schedl 2008), semantic music search engines (Knees et al. 2007), and intelligent user interfaces (Knees et al. 2007; Pampalk and Goto 2007) to access music collections by means more sophisticated than the textual browsing facilities (`artist-album-track` hierarchy) traditionally offered.

Various approaches to model the term vector space (Salton et al. 1975) on the Web have been proposed throughout the last years, e.g., Debole and Sebastiani (2003), Lan et al. (2005), Salton and Buckley (1988), Schedl et al. (2011), Whitman and Lawrence (2002). Microblogs, in contrast, have been studied to a much smaller extent, although using this data source for the purpose of similarity estimation between entities offers several advantages over the use of Web pages. First, microblog posts are shorter and typically more precise than Web pages, the former reducing computational complexity, the latter potentially offering more accurate results. Second, due to the instantaneous nature of microblogs, text-based similarity estimation approaches leveraging this kind of data are better capable of incorporating breaking news and offering a more up-to-date view on events related to the investigated domains, such as information on album releases or latest gossip about musicians or actors.

Addressing the lack of literature on modeling named entities via term vectors on the microblogosphere and thoroughly investigating different aspects of the models, the work at hand is the first aiming to answer the following research questions. First, we would like to assess if microblog data gathered over several months are capable of reflecting similarities between named entities from two domains, namely music artists and movies. We chose these two domains because accurate similarity measures are of particular importance in these contexts, which is underlined by the recent popularity and developments of recommender systems for music and movies, such as those offered by `last.fm` and `Netflix`, cf. Celma (2008), Koren (2009). The second important question that is addressed in this work is how to model similarities between the entities of interest. There exists a large number of possibilities to construct term vectors from texts/microblogs related to the

[1] http://twitter.com. Accessed January 2011.

named entities under consideration (in regard to term selection, term weighting, or normalization, for example). The corresponding algorithmic choices, together with the actual similarity measure employed, have a great impact on the accuracy of the similarity estimates between the music or movie entities. The objective of this work is hence to identify well-performing combinations of these choices and to derive general rules for modeling similarities between named entities from microblogs. Performance is measured by an evaluation approach resembling (Sanderson and Zobel 2005). More precisely, Mean Average Precision (MAP) scores are computed on genre labels predicted by a k-Nearest Neighbor (kNN) classifier. To reduce the computational complexity of evaluating the otherwise enormous set of different algorithmic combinations, results are first computed on a smaller set and only combinations statistically insignificantly different from the top-performing combination will be assessed on the larger data sets.

The work at hand was inspired by Zobel and Moffat (1998), where the authors thoroughly evaluate various choices related to constructing text feature vectors for IR purposes, e.g., term frequency (*TF*), term weights (*IDF*), and normalization approaches. They analyze the influence of these decisions on retrieval behavior. Similarly, a systematic large-scale study (in terms of single evaluation experiments and factors analyzed) on the influence of a multitude of decisions on similarity estimation, using real-world data collections, is presented here. To this end, we investigate several thousand combinations of the following single aspects:

- query scheme
- index term set
- term frequency
- inverse document frequency
- normalization with respect to document length
- similarity function

The *term frequency* $r_{d,t}$ of a term t in a document d estimates the importance t has for document d (representing the named entity under consideration). The *inverse document frequency* w_t estimates the overall importance of term t in the whole corpus and is commonly used to weight the $r_{d,t}$ factor, i.e., downweight terms that are important for many documents and hence less discriminative for d. We further assess the impact of *normalization* with respect to document length. Moreover, different *similarity functions* S_{d_1,d_2} to estimate the proximity between the term vectors of two named entities' documents d_1 and d_2 are examined.

The remainder of this article is organized as follows. Section 2 outlines the context of this work by conducting a literature review on text-based similarity measurement and microblog mining. Section 3 then describes all aspects we analyzed to model the named entity similarity space on the microblogosphere. The core part of this contribution can be found in Section 4, where details on the experiments are given and results are presented and discussed. Finally, conclusions are drawn in Section 5.

2 Related work

Related work basically falls into two categories: text-based similarity measurement and microblog mining. Whereas the former has a long tradition, ranging back several decades, the latter is a rather young research field.

2.1 Text-based similarity measures

There exists a wide range of literature on modeling text documents according to the bag-of-words principle using vector space representations, e.g., Baeza-Yates and Ribeiro-Neto (2011), Luhn (1957), Salton et al. (1975). Since elaborating on all publications related to the discipline of text-IR is out of this article's scope, we restrict ourselves to point to some work dealing with text-IR in the context of multimedia retrieval on the Web, as this context is closely related to the sets of named entities we use in the evaluation experiments.

Text data in the multimedia domain generally constitutes *context information* or *contextual data*, opposed to content-based features directly extracted from the media items. Deriving term feature vectors from Web pages for the purpose of music artist similarity calculation was first undertaken in Cohen and Fan (2000). Cohen and Fan automatically extract lists of artist names from Web pages, which are found by querying Web search engines. The resulting pages are then parsed according to their DOM tree, and all plain text content with minimum length of 250 characters is further analyzed for occurrences of entity names. Term vectors of co-occurring artist names are then used for artist recommendation. Using artist names to build term vector representations, whose term weights are computed as co-occurrence scores, is an approach also followed later in Schedl et al. (2005), Zadel and Fujinaga (2004). In contrast to Cohen and Fan's approach, the authors of Schedl et al. (2005), Zadel and Fujinaga (2004) derive the term weights from search engine's page count estimates and suggest their method for artist recommendation.

Automatically querying a Web search engine to determine pages related to a specific topic is a common and intuitive task, which is therefore frequently performed for data acquisition in IE research. Examples in the music domain can be found in Geleijnse and Korst (2006), Whitman and Lawrence (2002), whereas Cimiano et al. (2004), Cimiano and Staab (2004), Knees et al. (2007) apply this technique in a more general context.

Building term feature vectors from term sets other than artist names is performed in Whitman and Lawrence (2002), where Whitman and Lawrence extract different term sets (unigrams, bigrams, noun phrases, artist names, and adjectives) from up to 50 artist-related Web pages obtained via a search engine. After downloading the pages, the authors apply parsers and a part-of-speech (POS) tagger (Brill 1992) to assign each word to its suited test set(s). An individual term profile for each artist is then created by employing a version of the $TF \cdot IDF$ measure. The overlap between the term profiles of two artists, i.e., the sum of weights of all terms that occur in both term profiles, is then used as an estimate for their similarity.

Extending the work presented in Whitman and Lawrence (2002), Baumann and Hummel (2003) introduce filters to prune the set of retrieved Web pages. First, they remove all Web pages with a size of more than 40 kilobytes (after parsing). They also try to filter out advertisements by ignoring text in table cells comprising more than 60 characters, but not forming a correct sentence. Finally, Baumann and Hummel perform keyword spotting in the URL, the title, and the first text part of each page. Each occurrence of the initial query parts (artist name, "music", and "review") contributes to a page score. Pages that score too low are filtered out.

Knees et al.'s (2004) approach is similar to Whitman and Lawrence (2002). Unlike Whitman and Lawrence who experiment with different term sets, Knees et al. use only one list of unigrams. For each artist, a weighted term profile is created by applying a $TF \cdot IDF$ variant. Calculating the similarity between the term profiles of two artists is then performed using the cosine similarity. Knees et al. evaluate their approach in a genre classification

setting using as classifiers k-Nearest Neighbor (kNN) and Support Vector Machines (SVM) (Vapnik 1995).

Other approaches derive term profiles from more specific Web resources. In Celma et al. (2006), for example, the authors propose a music search engine that crawls audio blogs via RSS feeds and calculates $TF \cdot IDF$ features. Hu et al. (2005) extract TF-based features from music reviews gathered from `Epinions.com`.[2] In Schedl (2010) the author extracts user posts associated with music artists from the microblogging service `Twitter`[3] and models term profiles using term lists specific to the music domain.

In the work reported on so far, the authors usually select a specific variant of the $TF \cdot IDF$ term weighting measure and apply it to documents retrieved for the entity under consideration. The individual choices involved in selecting a specific $TF \cdot IDF$ variant and similarity function, however, do not seem to be the result of detailed assessments. They rather resemble common variants that are known to yield good results in IR tasks. Whether these variants are also suited to describe named entities via term profiles and subsequently estimate similarities between them is seldom assessed comprehensively in the literature. Sebastiani (2002) presents a review of different approaches to text categorization from a machine learning perspective, focusing on term selection techniques. Salton and Buckley (1988) investigate different approaches to term weighting and similarity measurement for text retrieval. Closest to the work at hand is certainly Zobel and Moffat's thorough study on various choices in modeling term profiles (Zobel and Moffat 1998). In particular, term weights for queries and documents as well as similarity functions are analyzed. However, Zobel and Moffat aim at determining good algorithmic choices for the purpose of document retrieval, i.e., retrieving relevant documents for a given query. We are, in contrast, interested in similarity measurement between two documents that represent named entities. Therefore, this article presents the first comprehensive study on named entity similarity estimation on the microblogosphere.

2.2 Microblog mining

With the advent of microblogging a huge, albeit noisy data source became available. Literature dealing with microblogs can be broadly categorized into works that study human factors or properties of the Twittersphere and works that exploit microblogs for information extraction and retrieval tasks.

As for the former, Teevan et al. (2011) analyze query logs to uncover differences in search behavior between users of classical Web search engines and users looking for information in microblogs. They found that `Twitter` queries are shorter and more popular than `bing`[4] queries on average. Furthermore, microblogs are more often sought for people, opinions, and breaking news. In terms of query formulation, reissuing the same query can be more frequently observed in microblog search. In Web search, by contrast, modifying and extending a query is very popular.

Java et al. (2007) study network properties of the microblogosphere as well as geographical distributions and intentions of `Twitter` users. The authors report that `Twitter` is most popular in North America, Europe, and Asia (Japan), and that same language is an important factor for cross-connections ("followers" and "friends") over continents. Employing the *HITS* algorithm (Kleinberg 1999) on the network of

[2] http://www.epinions.com/music. Accessed August 2007.

[3] See the footnote 1.

[4] http://www.bing.com. Accessed January 2010.

"friend"-relations, Java et al. further derived user intentions from structural properties. They identified the following categories: information sharing, information seeking, and friendship-wise relationships. Analyzing the content of Twitter posts, the authors distilled the following intentions: daily chatter, conversations, sharing information/URLs, and reporting news.

In a recent study, Kwak et al. (2010) perform a topological analysis of the Twitter network. The authors report a low level of reciprocity, i.e., only 22% of the connections between users are bidirectional. The average path length was found to be only four, which is surprisingly small for a network the size of the Twittersphere and considering the directional network structure. Moreover, a moderate level of homophily, i.e., a higher likelihood for connections between similar people than between dissimilar people, was discovered when measuring similarity in terms of geographic location and user popularity. In addition, Kwak et al.'s study indicates that information diffusion after the first retweet is very fast.

Work related to content mining of microblogs includes the following: Cheng et al. propose a method to localize Twitter users based on spatial cues ("local" words) extracted from their tweets' content (Cheng et al. 2010). To this end, in a first step several classifiers are trained to identify words with a strong geospatial meaning. In order to deal with the sparsity in the distribution of these cues, different smoothing approaches, e.g., taking into account neighboring cities when constructing the term representation of a city, are applied subsequently. In an experiment conducted on a set of tweets posted within the USA, Cheng et al.'s approach placed more than a half of the users within a 100-mile-radius of their correct location.

Making use of the fact that tweets are a good source for up-to-date information and breaking news, Dong et al. (2010) propose an approach to identify fresh URLs in Twitter posts. To this end, the authors investigate content-based features extracted from the tweets, an authority score computed for each user, and Twitter-specific statistical features, such as number of retweets or number of users that replied to a message containing a tiny URL. They show that these features can be used to improve both recency ranking and relevance ranking in real-time Web search. Another work that aims at improving ranking can be found in Duan et al. (2010). Duan et al. propose a novel ranking strategy for tweet retrieval. To this end, they investigate different feature sets, including content-based features, Twitter-specific features, and authority scores of users (followers, retweeters, mentioners). Using a learning to rank algorithm, the authors found that the best-performing features are authority scores, length of a tweet, and whether the tweet contains a URL.

An approach to classifying tweets can be found in Sriram et al. (2010). Sriram et al. describe each tweet by an eight-dimensional feature vector comprising the author of the post and seven binary attributes indicating, for example, occurrence of slang words, currency and percentage signs, or the use of capitalization and repeated characters. Sriram et al.'s feature set outperformed the standard bag-of-words approach using a Naïve Bayes classifier to categorize tweets into the five classes news, events, opinions, deals, and private messages.

Armentano et al. (2011) present a recommender system that suggests potentially interesting users to follow based on the similarity between tweets posted by the seed user and tweets posted by a set of candidate users. To this end, the authors create and investigate different user profiles, for example, modeling the seed user via term frequencies of his/her aggregate posts or of all of his/her followees. Related to Armentano et al.'s work, Weng et al. aim at identifying influential twitterers for a given topic (Weng et al. 2010). To this

end, they apply *Latent Dirichlet Allocation* (LDA) (Blei et al. 2003) to their corpus of tweets. Subsequently, topical similarity between twitterers is computed as the Jensen–Shannon divergence between the distribution of the latent topics of the respective users. Further taking into account the link structure, Weng et al. propose a ranking function for influential twitterers in each topic. Similar to Armentano et al. (2011), Weng et al. evaluate their approach in a recommendation setting.

Microblogs have also been exploited for the purpose of event and trend detection. Sakaki et al. propose semantic analysis of tweets to detect earthquakes in Japan in real-time (Sakaki et al. 2010). A more general approach to automatically detect events and summarize trends by analyzing tweets is presented by Sharifi et al. (2010). Another work on trend detection is Schedl (2011), where Schedl exploits tweets for spatio-temporal popularity estimation of music artists. Sankaranarayanan et al. aim at capturing tweets that report on breaking news (Sankaranarayanan et al. 2009). They cluster the identified tweets according to their $TF \cdot IDF$ weights and cosine similarity. Furthermore, each cluster is assigned a set of geographic locations using both spatial clues in the tweets themselves and explicit location information as indicated by the twitterers.

3 Modeling the microblog term vector space

Resembling the large-scale experiments conducted in Zobel and Moffat (1998), our analysis is guided by the question whether specific algorithmic choices perform consistently and considerably better or worse than others. Performance is measured via classification tasks among term vector representations of tweets, cf. Sect. 4. Our goal is, hence, to derive guidelines for favoring or avoiding specific algorithmic variants when the task is similarity estimation between named entities and the corpus comprises microblogs. The assessed aspects for modeling named entities based on microblogs are detailed in the following (Table 1).

3.1 Query scheme

We decided to assess two different schemes to query `Twitter` as previous work on Web-based IE (Schedl et al. 2005; Whitman and Lawrence 2002) has shown that adding domain-specific key terms to a search request generally improves the quality of feature vectors in terms of similarity-based classification accuracy. In Web-based music

Table 1 Denominations used in term weighting functions and similarity measures

$\mathcal{D}$	Set of documents
N	Number of documents
$f_{d,t}$	Number of occurrences of term t in document d
f_t	Number of documents containing term t
F_t	Total number of occurrences of t in the collection
$\mathcal{T}_d$	Set of distinct terms in document d
f_d^m	Largest $f_{d,t}$ of all terms t in d
f^m	Largest f_t in the collection
$r_{d,t}$	Term frequency (cf. Table 5)
w_t	Inverse document frequency (cf. Table 6)
W_d	Document length of d

Table 2 Query schemes used to retrieve music/movie-related tweets

Abbr.	Query scheme
QS_A	"artist name" / "movie name"
QS_M	"artist name"+music / "movie name"+movie

information research, for example, common terms used as additional key words are "music review" or "music genre style". Taking into account the 140-character-limitation of tweets, we decided to include only "music" as additional query term (QS_M) for the music data sets, or we query without any additional key terms, i.e., use only the artist name (QS_A). For the movie data set, the setting QS_M refers to including the term "movie" in the query. Table 2 summarizes the two query schemes investigated.

3.2 Index term set

Earlier work in text-based music artist modeling (Turnbull et al. 2007; Hu and Downie 2007; Pampalk et al. 2005) shows that a crucial choice in defining the representation of an artist is that of the terms used to index the corresponding documents. For the work at hand, we hence investigated various term sets, which are summarized for the music and movie collections, respectively, in Tables 3 and 4. Set TS_A contains all terms found in the corpus (after casefolding, stopping, and stemming). Set TS_S is the entire term dictionary of SCOWL,[5] which is an aggregation of several spell checker dictionaries for various English languages and dialects. Set TS_N encompasses all artist names present in the music data set. Previous work has shown that the corresponding *co-occurrence* approach to music artist similarity estimation yields remarkable results (Schedl and Knees 2008; Schedl et al. 2005). Term set TS_D is a manually created dictionary of music-related terms that resembles the one used in Pampalk et al. (2005). It contains, for example, descriptors of genre, instruments, geographic locations, epochs, moods, and musicological terms. Set TS_L represents the 250 most popular tags utilized by users of last.fm. Set TS_F comprises the aggregated data sets for the data types *musical genre*, *musical instrument*, and *emotion*, extracted from Freebase.[6]

For the movie data set (cf. Table 4), we adapted the term sets accordingly. Sets TS_A and TS_S conceptually equal the corresponding sets used to index music-related tweets. Term set TS_D, in contrast, is a dictionary of movie-related terms, which we extracted from the "key words" provided by IMDb. Since this key word set is considerably noisy, we performed frequency-based filtering. We retained only terms that were assigned to at least 10 different movies, but to not more than 100 different movies. The former constraint effectively removes noise, the latter discards terms that are unlikely to discriminate well between different categories of movies.

To build the inverted word-level index (Zobel and Moffat 2006), we use a modified version of the open source indexer Lucene,[7] which we extended to represent Twitter posts. The extensions will be made available through the CoMIRVA framework[8] (Schedl et al. 2007). When creating the indexes for the different term sets, we commonly employ casefolding and stopping, e.g., Baeza-Yates and Ribeiro-Neto (2011). Stemming, in

[5] http://wordlist.sourceforge.net. Accessed January 2011.

[6] http://www.freebase.com. Accessed January 2011.

[7] http://lucene.apache.org. Accessed January 2011.

[8] http://www.cp.jku.at/CoMIRVA. Accessed January 2011.

Table 3 Different term sets used to index the music-related Twitter posts

Abbr./term set	Cardinality	Description
TS_A/all_terms	C224a, QS_A: 38,133 C224a, QS_M: 19,133 C3ka, QS_A: 1,489,459 C3ka, QS_M: 437,014	All terms (stemmed) that occur in the corpus of the retrieved Twitter posts
TS_S/scowl_dict	698,812	All terms that occur in the entire SCOWL dictionary
TS_N/artist_names	224/3,000	Names of the artists for which data was retrieved
TS_D/dictionary	1,398	Manually created dictionary of musically relevant terms
TS_L/last.fm_toptags	250	Overall top-ranked tags returned by last.fm's *Tags.getTopTags* function
TS_F/freebase	3,628	Music-related terms extracted from Freebase (genres, instruments, emotions)

Table 4 Different term sets used to index the movie-related Twitter posts

Abbr./term set	Cardinality	Description
TS_A/all_terms	QS_A: 1,843,286/54,378 QS_M: 754,067/29,532	All terms (stemmed) that occur in the corpus of the retrieved Twitter posts
TS_S/scowl_dict	QS_A: 698,812/28,355 QS_M: 698,812/12,473	All terms that occur in the entire SCOWL dictionary
TS_D/dictionary	QS_A: 25,527/4,877 QS_M: 25,527/3,569	Dictionary of filtered IMdb key words

contrast, is only performed for the term sets for which it seems reasonable, i.e., for term sets TS_A and TS_S.

3.3 TF and IDF: term weighting

The term weighting models investigated here resemble Zobel and Moffat's (1998). We decided to extend the $TF \cdot IDF$ formulations investigated by them with *BM25*-like formulations. The assessed variants for *TF* can be found in Table 5, those for *IDF* are shown in Table 6. Table 1 contains an overview of the denominations used in the different term weighting formulations, normalization strategies, and similarity measures (Tables 7, 8).

BM25 is an alternative term weighting scheme, used in the Okapi framework for text-based probabilistic retrieval, cf. Robertson et al. (1995, 1999). This model assumes a priori knowledge on topics from which different queries are derived. Moreover, based on information about which documents are relevant for a specific topic and which are not, the term weighting function can be tuned to the corpus under consideration. Since *BM25* is a well-established term ranking method, we included it in the experiments. However, it has to be noted that we cannot assume categorical a priori knowledge here, neither on the level of single tweets, nor on the level of named entities. On the level of tweets, manually classifying hundreds of thousands of posts would be too labor-intensive. On the named entity level, we could obviously group the entities (or more precisely, the corresponding

Table 5 Evaluated variants to calculate the term frequency $r_{d,t}$

Abbr.	Description	Formulation
TF_A	Formulation used for binary match SB = b	$r_{d,t} = \begin{cases} 1 & \text{if } t \in \mathcal{T}_d \\ 0 & \text{otherwise} \end{cases}$
TF_B	Standard formulation SB = t	$r_{d,t} = f_{d,t}$
TF_C	Logarithmic formulation	$r_{d,t} = 1 + \log_e f_{d,t}$
TF_C2	Alternative logarithmic formulation suited for $f_{d,t} < 1$	$r_{d,t} = \log_e (1 + f_{d,t})$
TF_C3	Alternative logarithmic formulation as used in *ltc* variant	$r_{d,t} = 1 + \log_2 f_{d,t}$
TF_D	Normalized formulation	$r_{d,t} = \frac{f_{d,t}}{f_d^m}$
TF_E	Alternative normalized formulation. Similar to Zobel and Moffat (1998) we use $K = 0.5$. SB = n	$r_{d,t} = K + (1 - K) \cdot \frac{f_{d,t}}{f_d^m}$
TF_F	Okapi formulation, according to Robertson et al. (1995), Zobel and Moffat (1998). For W we use the vector space formulation, i.e., the Euclidean length	$r_{d,t} = \frac{f_{d,t}}{f_{d,t} + W_d / av_{d \in D}(W_d)}$
TF_G	Okapi BM25 formulation, according to Robertson et al. (1999)	$r_{d,t} = \frac{(k_1+1) \cdot f_{d,t}}{f_{d,t} + k_1 \cdot \left[(1-b) + b \cdot \frac{w_d}{av_{d \in D}(W_d)} \right]}$ $k_1 = 1.2,\ b = 0.75$

Table 6 Evaluated variants to calculate the inverse document frequency w_t

Abbr.	Description	Formulation
IDF_A	Formulation used for binary match SB = x	$w_t = 1$
IDF_B	Logarithmic formulation SB = f	$w_t = \log_e \left(1 + \frac{N}{f_t}\right)$
IDF_B2	Logarithmic formulation used in *ltc* variant	$w_t = \log_e \left(\frac{N}{f_t}\right)$
IDF_C	Hyperbolic formulation	$w_t = \frac{1}{f_t}$
IDF_D	Normalized formulation	$w_t = \log_e \left(1 + \frac{f_m}{f_t}\right)$
IDF_E	Another normalized formulation SB = p	$w_t = \log_e \frac{N - f_t}{f_t}$
	The following definitions are based on the term's noise n_t and signal s_t.	$n_t = \sum_{d \in \mathcal{D}_t} \left(-\frac{f_{d,t}}{F_t} \log_2 \frac{f_{d,t}}{F_t} \right)$ $s_t = \log_2 (F_t - n_t)$
IDF_F	Signal	$w_t = s_t$
IDF_G	Signal-to-noise ratio	$w_t = \frac{s_t}{n_t}$
IDF_H		$w_t = \left(\max_{t' \in \mathcal{T}} n_{t'} \right) - n_t$
IDF_I	Entropy measure	$w_t = 1 - \frac{n_t}{\log_2 N}$
IDF_J	Okapi BM25 IDF formulation, according to Pérez-Iglesias et al. (2009), Robertson et al. (1999)	$w_t = \log \frac{N - f_t + 0.5}{f_t + 0.5}$

tweets) according to a genre taxonomy and optimize *BM25* correspondingly. However, we believe that this is not justifiable for two reasons: First, for arbitrary media repositories, we cannot assume to have access to genre information. Second, using genre information would obviously bias the results of the genre classification experiments as the other term weighting measures do not incorporate such a priori knowledge. Thus, *BM25* would be

Table 7 Evaluated normalization strategies for document length

Abbr.	Description	Formulation
NORM_NO	No normalization	
NORM_SUM	Normalize sum of each virtual document's term feature vector to 1	$\sum_{t \in \mathcal{T}_d} r_{d,t} = 1$
NORM_MAX	Normalize maximum of each virtual document's term feature vector to 1	$\max_{t \in \mathcal{T}_d} r_{d,t} = 1$

Table 8 Evaluated similarity functions S_{d_1,d_2}

Abbr.	Description	Formulation
SIM_INN	Inner product	$S_{d_1,d_2} = \sum_{t \in \mathcal{T}_{d_1,d_2}} \left(w_{d_1,t} \cdot w_{d_2,t}\right)$
SIM_COS	Cosine measure	$S_{d_1,d_2} = \frac{\sum_{t \in \mathcal{T}_{d_1,d_2}} \left(w_{d_1,t} \cdot w_{d_2,t}\right)}{W_{d_1} \cdot W_{d_2}}$
SIM_DIC	Dice formulation	$S_{d_1,d_2} = \frac{2 \sum_{t \in \mathcal{T}_{d_1,d_2}} \left(w_{d_1,t} \cdot w_{d_2,t}\right)}{W_{d_1}^2 + W_{d_2}^2}$
SIM_JAC	Jaccard formulation	$S_{d_1,d_2} = \frac{\sum_{t \in \mathcal{T}_{d_1,d_2}} \left(w_{d_1,t} \cdot w_{d_2,t}\right)}{W_{d_1}^2 + W_{d_2}^2 - \sum_{t \in \mathcal{T}_{d_1,d_2}} \left(w_{d_1,t} \cdot w_{d_2,t}\right)}$
SIM_OVL	Overlap formulation	$S_{d_1,d_2} = \frac{\sum_{t \in \mathcal{T}_{d_1,d_2}} \left(w_{d_1,t} \cdot w_{d_2,t}\right)}{\min(W_{d_1}^2, W_{d_2}^2)}$
SIM_EUC	Euclidean similarity	$D_{d_1,d_2} = \sqrt{\sum_{t \in \mathcal{T}_{d_1,d_2}} \left(w_{d_1,t} - w_{d_2,t}\right)^2}$
		$S_{d_1,d_2} = \left(\max_{d_1',d_2'} (D_{d_1',d_2'})\right) - D_{d_1,d_2}$
SIM_JEF	Jeffrey divergence-based similarity	$S_{d_1,d_2} = \left(\max_{d_1',d_2'} (D_{d_1',d_2'})\right) - D_{d_1,d_2}$
		$D(F,G) = \sum_i \left(f_i \log \frac{f_i}{m_i} + g_i \log \frac{g_i}{m_i}\right)$
		$m_i = \frac{f_i + g_i}{2}$

unjustifiably favored. For our experiments, we therefore use a simpler *BM25* formulation as the one proposed in Robertson et al. (1999), cf. variants TF_G and IDF_J in Tables 5 and 6, respectively.

3.4 Virtual documents and normalization

When creating a Web-based term profile that describes a named entity (a music artist or movie in our case), it is common to aggregate the Web pages associated with the entity under consideration to form a "virtual document" (Baumann and Hummel 2003; Knees et al. 2004). This procedure not only facilitates handling small or empty pages, it is also more intuitive since the item of interest is the entity under consideration, not a Web page. The study conducted in Schedl et al. (2011) further shows that calculating term weights on the level of individual Web pages before aggregating the resulting feature vector performs inferior for the task of similarity calculation than using "virtual documents". Therefore it seems reasonable to aggregate all tweets retrieved for a named entity to one "virtual post",

in particular taking into consideration the already strong limit of Twitter posts to 140 characters.

Since the different length of two entity's virtual documents might influence the performance of retrieval and similarity prediction tasks, e.g., Baeza-Yates and Ribeiro-Neto (2011), we evaluate several normalization methods, which are summarized in Table 7.

3.5 Similarity function

The similarity measures analyzed are shown in Table 8. We included all measures investigated by Zobel and Moffat (1998) that can be applied to our somewhat differing usage scenario of computing similarities between two equally dimensional term feature vectors that represent two comparable named entities. In addition, Euclidean similarity (SIM_EUC) and similarity inferred from Jeffrey divergence (SIM_JEF) (Lin 1991) were included.

3.6 Notation

To facilitate referring to a particular evaluation experiment, which is defined as a combination of the choices described above, we adopt the following scheme to denote one algorithmic setting:

```
<Query Scheme>.<Index Term Set>.<Normalization>.
    <TF>.<IDF>.<Similarity Measure>
```

Omitting certain components, we denote sets of algorithmic combinations: e.g., TF_C.IDF_B.SIM_COS refers to all experiments with term frequency formulation TF_C, inverse document formulation IDF_B, and the cosine similarity function, irrespective of query scheme, index term set, and document normalization.

4 Evaluation

4.1 Data sets

We performed evaluation using three data sets, covering two types of named entities that relate to two different media types: *music artists* and *movie titles*. The creation of these data sets is outlined and their properties are presented in the following.

4.1.1 Music artists

We used two data sets of music artists for evaluation. The first one, referred to as C224a, consists of 224 well-known artists and has a uniform genre distribution (14 genres,[9] 16 artists each). It has been frequently used to evaluate Web-/text-based music information retrieval approaches.[10]

[9] The genres in C224a are Country, Folk, Jazz, Blues, R'n'B/Soul, Heavy Metal/Hard Rock, Alternative Rock/Indie Punk, Rap/Hip Hop, Electronica, Reggae, Rock'n'Roll, Pop, and Classical.

[10] C224a is available at http://www.cp.jku.at/people/schedl/data/C224a.txt.

Table 9 Genre distribution of music artist set C3ka

Genre	Artists	Share (%)
Avantgarde	8	0.267
Blues	11	0.367
Celtic	5	0.167
Classical	42	1.400
Country	24	0.800
Easy listening	6	0.200
Electronica	149	4.967
Folk	24	0.800
Gospel	23	0.767
Jazz	106	3.533
Latin	91	3.033
Newage	18	0.600
Rap	203	6.767
Reggae	29	0.967
RnB	101	3.367
Rock	2,031	67.700
Vocal	30	1.000
World	99	3.300

The second data set consists of 3,000 music artists, representing a large real-world collection. The data has been gathered as follows. We used last.fm's API[11] to extract the most popular artists for each country of the world, which we then aggregated into a single list of 201,135 unique artist names. Since last.fm's data is prone to misspellings or other mistakes due to its collaborative, user-generated knowledge base, we cleaned the data set by matching each artist name with the database of the expert-based music information system allmusic.com,[12] from which we also extracted genre information. Starting this matching process from the most popular artist found by last.fm and including only artist names that also occur in allmusic.com, we eventually obtained a list of 3,000 music artists. This artist set, which will be denoted C3ka in the following, is publicly available.[13] According to allmusic.com the artists are categorized into 18 distinct genres. The distribution of the genres in C3ka is shown in Table 9. Please note that the editors of allmusic.com use the genre "Rock" to denote a widespread range of music; basically, everything from Pop to Dark Metal is classified as "Rock". Therefore, the genre distribution is considerably unbalanced.

4.1.2 Movies

The second data set consists of 1,008 distinct movie titles extracted from IMDb (Jass 2003). For 25 movie genres, we gathered the 50 top-ranked movies. We further added the overall 50 top-ranked movies of each decade, from the 1910s to the 2010s. This adds a

[11] http://last.fm/api. Accessed January 2011.

[12] http://www.allmusic.com. Accessed January 2011.

[13] C3ka is available at http://www.cp.jku.at/people/schedl/data/C3ka.txt.

further 11 categories. Please note that some movies occur in the top-ranked list for more than one genre, hence the total number of 1,008 distinct movie titles. The movie data set will be referred to as C1km in the following, and the movie names are available for download.[14]

4.2 Acquiring tweets

To gather posts related to the two domains under assessment, i.e., music and movies, we use Twitter's API[15] to issue queries according to the schemes indicated in Table 2. Accounting for the time-varying behavior of the search results and to obtain a broad coverage, we queried Twitter from December 2010 to February 2011 and aggregated the posts retrieved over time for each query. The resulting set of tweets per query/named entity is then pre-processed by employing casefolding and stopping. When using the term sets TS_A and TS_S, stemming is employed additionally.

For artist set C224a, we achieved a coverage of 100%; for set C3ka, we achieved a coverage of 96.87%, i.e., for 2,906 artists out of the 3,000 tweets were available. Coverage for the movie data set C1km was considerably lower (82.8% or 834 movies), likely due to the fact that IMDb always lists the full, official movie title, which is often replaced by a shortened version when referring to the movie in a microblog, e.g., "The Fog of War: Eleven Lessons from the Life of Robert S. McNamara".

As for the total amounts of tweets extracted, using collection C224a, 21,336 tweets were gathered for QS_A and 10,867 for QS_M. For set C3ka, 3,161,582 tweets were retrieved for QS_A and 2,972,130 for QS_M. For the movie set C1km, we retrieved 11,684,074 tweets using query scheme QS_A and 4,958,223 tweets using query scheme QS_M.

4.3 Experimental setup

To assess the quality of the named entity's term models, we perform *genre classification* experiments, evaluating the different algorithmic choices. As ground truth the genre labels given by allmusic.com and IMDb are used for the music sets and the movie set, respectively. Although genre taxonomies are often inconsistent and erroneous (Pachet and Cazaly 2000), it has become commonplace to use genre as a proxy for similarity. In principle, a more precise ground truth could be established from human similarity judgments. Complete similarity judgments are, however, not publicly available on a large scale, neither for music, nor for movies. Hence, we have to restrict evaluation to the retrieval task of determining k artists/movies similar to a given query artist/movie. This task resembles k nearest neighbor (kNN) classification, where the class of a seed item is predicted as the most frequent class among the seed's k most similar items. In the case of the single-class classification problem given by the music data sets, performing kNN is straightforward. However, when dealing with multiple labels/classes assigned to each item, as in the case of the movie set, we opted to employ a strict decision rule: Given a seed item with s class labels associated and a number of k nearest neighbors to consider, we accumulate the number of occurrences of up to s classes among the k neighbors. We then calculate the (proportionate) precision of the top s classes given by the accumulated counts on the seed's s classes, i.e., each of the top s classes among the k nearest neighbors that match one of the

[14] C1km is available at http://www.cp.jku.at/people/schedl/data/C1km.txt.

[15] https://dev.twitter.com. Accessed February 2012.

seed's s classes account for a precision score of $1/s$. The algorithm used to compute precision@k for the multi-class experiments is illustrated in Algorithm 1.

We performed a two-staged evaluation: In order to determine and filter inferior algorithmic combinations, we first ran a comprehensive set of evaluation experiments on the equally genre-distributed data set `C224a`. In a second set of experiments, we then evaluated the remaining variants on the real-world artist set `C3ka`. On the movie set `C1km` all variants were evaluated.

Our experimental setting resembles the ones employed in Buckley and Voorhees (2000), Sanderson and Zobel (2005). Given a query item, the retrieval task is to find items of the same class(es) via similarity. We use *Mean Average Precision* (MAP) as performance measure. Employing Algorithm 1, MAP is simply computed as the arithmetic mean of the precision@k scores. Following Sanderson and Zobel (2005), we first calculate MAP of each distinct algorithmic setting on data set `C224a`. Excluding redundant combinations, a total of 23,100 single experiments have been conducted for set `C224a` and 11,627 for set `C1km`. In the first stage of the experiments, only variants that fulfill at least one of the following two conditions are retained:

- there is a relative MAP difference of 10% or less to the top-ranked variant
- or the t test does not show a significant difference to the top-ranked variant (at 5% significance level).

For set `C224a`, the top 577 variants have a relative MAP difference (from the 1st to the respective rank, taking the respective rank as basis) of less than 10%. The pairwise t test shows a significant difference for the top-ranked 1,809 variants. For the second stage of experimentation, conducted on collection `C3ka`, we therefore evaluated only these top-ranked 1,809 variants. For the movie set `C1km`, these numbers are 2,392 (relative MAP difference) and 5,629 (t test), respectively (Figs. 1, 2).

Algorithm 1 Precision@k in the multi-class case

$A \leftarrow$ set of all artists
$a \leftarrow$ seed artist to compute precision for
$NN(a) \leftarrow$ sort $(\|a - A_i\|)$
$NN_k(a) \leftarrow k^{\text{th}}$nearest neighbor of a
$C_a[1..S] \leftarrow$ S classes artist a belongs to
$K \leftarrow$ number of nearest neighbors to consider

{for all artists in A}
for $a : A$ **do**
  {for different k in K-NN experiments}
  for $k = 1$ to K **do**
    {precision for k^{th} nearest neighbor}
    $prec(a, k) \leftarrow \frac{|C_{NN_k(a)} \cap C_a|}{S}$
  end for
  {accumulate precisions among k nearest neighbors}
  $prec(a)@k \leftarrow \frac{1}{k} \cdot \sum_{l=1..k} prec(a, l)$
end for
{return average precision@k}
return $\frac{prec(a)@k}{|A|}$

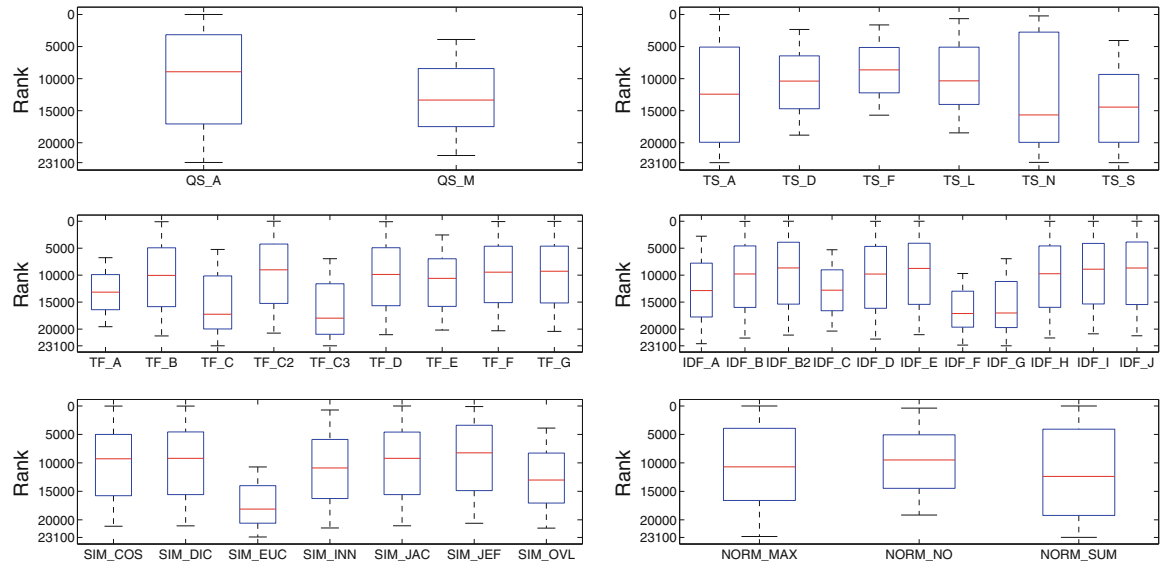


Fig. 1 Box plots of ranks for each algorithmic choice on music set `C224a`

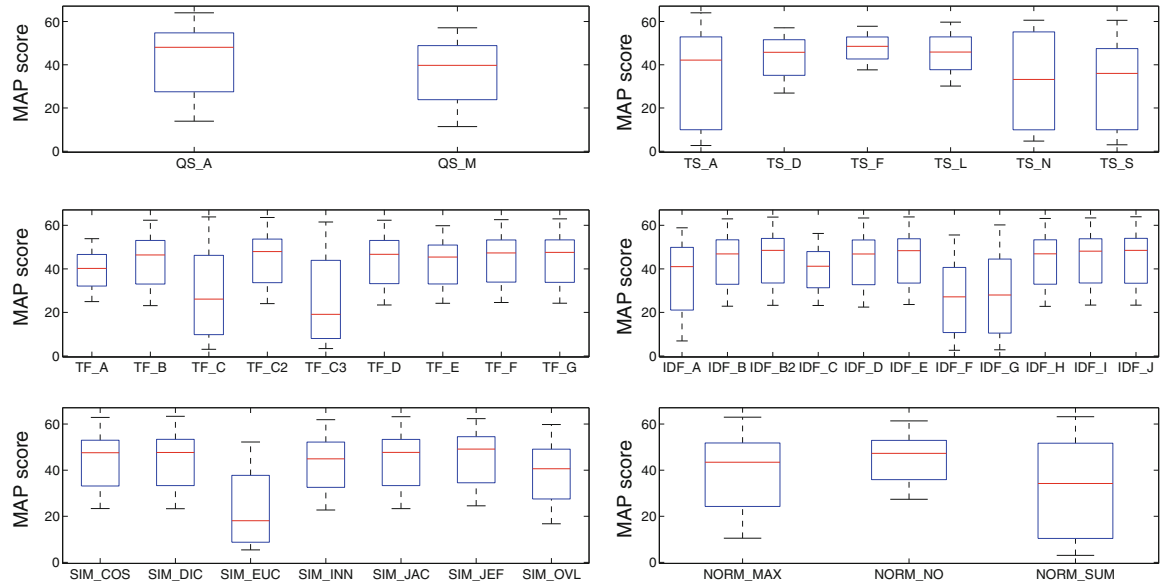


Fig. 2 Box plots of MAP scores for each algorithmic choice on music set `C224a`

4.4 Results and discussion

4.4.1 MAP scores

Table 10 shows the 10 top-ranked and the 10 bottom-ranked variants with their MAP scores (considering up to 15 nearest neighbors) for set `C224a`. The MAP scores of the 23,100 evaluated variants span a wide range and are quite diverse (cf. Fig. 3), with a mean of $\mu = 37.89$ and a standard deviation of $\sigma = 17.16$. From Table 10 it can be seen that highest MAP scores can only be achieved when using `QS_A`, `TS_A`, and `NORM_NO`. At the other end of the ranking we see that `QS_M` and `SIM_OVL` dominate the most inferior variants.

Table 11 shows the top- and bottom-ranked variants with their MAP scores for the movie data set `C1km` (considering up to 50 nearest neighbors). Note that these MAP scores

Table 10 MAP scores of the top-ranked and bottom-ranked variants on music set C224a

MAP	Variant
64.018	QS_A.TS_A.NORM_NO.TF_C2.IDF_E.SIM_JAC
63.929	QS_A.TS_A.NORM_NO.TF_C2.IDF_J.SIM_JAC
63.839	QS_A.TS_A.NORM_NO.TF_C.IDF_E.SIM_JAC
63.810	QS_A.TS_A.NORM_NO.TF_C2.IDF_E.SIM_COS
63.780	QS_A.TS_A.NORM_NO.TF_C.IDF_E.SIM_COS
63.780	QS_A.TS_A.NORM_NO.TF_C2.IDF_B2.SIM_JAC
63.780	QS_A.TS_A.NORM_NO.TF_C2.IDF_B2.SIM_DIC
63.720	QS_A.TS_A.NORM_NO.TF_C2.IDF_E.SIM_DIC
63.601	QS_A.TS_A.NORM_NO.TF_C2.IDF_J.SIM_COS
63.542	QS_A.TS_A.NORM_NO.TF_C.IDF_J.SIM_JAC
...	...
3.482	QS_M.TS_A.NORM_MAX.TF_G.IDF_G.SIM_OVL
3.452	QS_M.TS_S.NORM_SUM.TF_B.IDF_F.SIM_OVL
3.423	QS_M.TS_A.NORM_SUM.TF_C3.IDF_J.SIM_OVL
3.363	QS_M.TS_S.NORM_MAX.TF_G.IDF_F.SIM_OVL
3.274	QS_M.TS_A.NORM_SUM.TF_C.IDF_E.SIM_OVL
3.065	QS_M.TS_A.NORM_SUM.TF_C.IDF_J.SIM_OVL
3.006	QS_M.TS_A.NORM_MAX.TF_G.IDF_F.SIM_OVL
2.976	QS_M.TS_S.NORM_MAX.TF_F.IDF_F.SIM_OVL
2.857	QS_M.TS_A.NORM_MAX.TF_F.IDF_G.SIM_OVL
2.649	QS_M.TS_A.NORM_MAX.TF_F.IDF_F.SIM_OVL

are overall lower than the scores for the music collections, with a mean of $\mu = 23.12$ and a standard deviation of $\sigma = 2.61$. This lower overall performance is partly due to the higher number of classes, partly because of the stricter decision rule employed in the classification process, cf. Sect. 4.3. Highest ranks are again dominated by query scheme QS_A and term set TS_A, whereas the lowest-ranking variants are dominated by QS_A.TS_S.NORM_SUM.SIM_JEF.

When comparing Tables 10 and 11, it becomes obvious that the best- and worst-performing variants vary considerably with the set of names entities, in particular in terms of *TF* and *IDF* formulations as well as similarity measures. Furthermore, it seems easier to identify algorithmic choices that yield worse performance and should thus be avoided than to clearly suggest best-performing choices.

4.4.2 Distribution of specific algorithmic choices

Figure 4 displays the distribution of each analyzed aspect among all 23,100 experimental setups investigated for set C224a. Figure 5 shows this distribution among the 1,809 top-ranked variants. Figure 6 shows the top-ranked algorithmic choices for artist set C3ka and Fig. 7, eventually, shows this distribution for the movie data set C1km.

For some aspects, general rules can be derived from these plots: Regarding the query scheme, it is obvious that using only the named entity as indicator to determine related tweets (QS_A) outperforms adding domain-specific key words. This result at first glance contrasts earlier work on Web-based music artist classification (Knees et al. 2008).

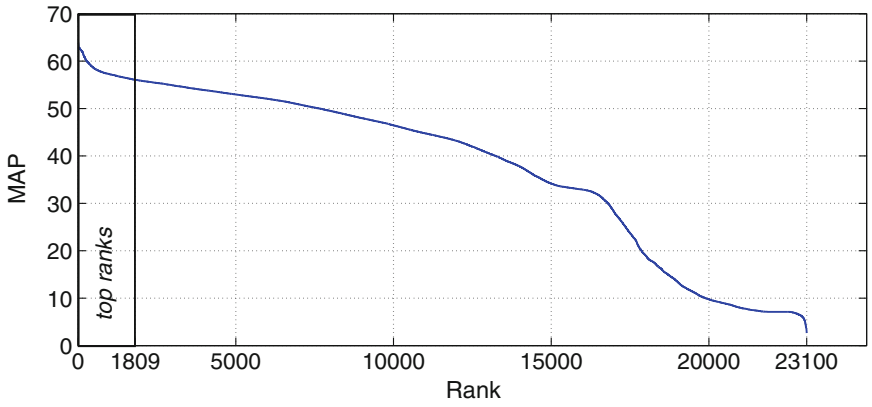


Fig. 3 Distribution of MAP scores among all 23,100 ranks on music set C224a

However, Knees et al. analyze Web pages, not microblogs. It seems that adding any additional key word too strongly prunes Twitter's result set.

As for the term sets used for indexing, the very top ranks are dominated by algorithmic variants that use the complete set of terms occurring in the corpus (TS_A), for both the music and the movie data sets. It is noteworthy, however, that the good performance of the general term sets (TS_A and TS_S) comes at the price of much higher computational complexity (cf. Tables 3, 4 for term set cardinalities). Hence, when performance is crucial, the results suggest using other term sets. A particularly good choice when the domain is music at first glance seems to be TS_N, the list of artist names, as it is the set that most frequently occurs among the top-ranked variants (32.5% or 588 times). However, TS_N yields very unstable results, as will be shown in the subsequent subsection. Another interesting finding is that the music dictionary TS_D, despite its good performance for similarity-based artist clustering using *Web pages*, cf. Pampalk et al. (2005), occurs first only at rank 1,112. An empirically verified reason for this may be that Twitter users tend to refrain from using a decent music-specific vocabulary, even when they twit about music-related issues.[16] For the movie set C1km, in contrast, TS_D represents a good trade-off between computational complexity and accuracy as it does not significantly more seldom occur among the top-ranked variants than the set TS_S (both about 28 vs. 44% for TS_A). It seems that a collaboratively assembled dictionary, such as TS_D for the movie domain, outperforms a domain-specific one assembled by experts, such as TS_D for the music domain, provided it is not too small.

As for the term weighting functions (*TF* and *IDF* variants), no clear picture regarding favorable variants emerges when analyzing the top-ranked algorithmic combinations. We found, however, that TF_A occurred most seldom among the top-ranked variants, regardless of the data set. This variant should thus be avoided. The most frequently occurring formulations on the other hand are TF_C2 (15.69% of the top-ranks for the music sets) and TF_E (16.80%), the latter being particularly present in the very top ranks for the music data sets. TF_C2 also occurs frequently among the top-ranked variants of the movie set C1km (13.52%), together with TF_D (14.55%), TF_F (13.82%), and TF_G (13.87%).

Analogously to *TF*, for *IDF* variants we can easily point to formulations that should be avoided, namely IDF_G (0.50% among C3ka's top ranks), IDF_F (0.66%), and IDF_A

[16] Only 478 unique terms out of the 1,398 in TS_D were used, only 319 were used in at least two different tweets.

Table 11 MAP scores of the top-ranked and bottom-ranked variants on movie set C1km

MAP	Variant
27.964	QS_A.TS_A.NORM_SUM.TF_G.IDF_C.SIM_INN
27.962	QS_A.TS_A.NORM_NO.TF_A.IDF_C.SIM_DIC
27.962	QS_A.TS_A.NORM_NO.TF_A.IDF_C.SIM_JAC
27.962	QS_A.TS_A.NORM_MAX.TF_A.IDF_C.SIM_DIC
27.962	QS_A.TS_A.NORM_MAX.TF_A.IDF_C.SIM_JAC
27.895	QS_A.TS_A.NORM_NO.TF_E.IDF_C.SIM_DIC
27.895	QS_A.TS_A.NORM_NO.TF_E.IDF_C.SIM_JAC
27.895	QS_A.TS_A.NORM_SUM.TF_E.IDF_C.SIM_DIC
27.895	QS_A.TS_A.NORM_SUM.TF_E.IDF_C.SIM_JAC
27.895	QS_A.TS_A.NORM_MAX.TF_E.IDF_C.SIM_DIC
...	...
17.101	QS_M.TS_S.NORM_SUM.TF_C3.IDF_H.SIM_JEF
17.101	QS_M.TS_S.NORM_SUM.TF_C3.IDF_I.SIM_JEF
17.101	QS_M.TS_S.NORM_SUM.TF_D.IDF_F.SIM_JEF
17.101	QS_M.TS_S.NORM_SUM.TF_D.IDF_G.SIM_JEF
17.101	QS_M.TS_S.NORM_SUM.TF_E.IDF_F.SIM_JEF
17.101	QS_M.TS_S.NORM_SUM.TF_E.IDF_G.SIM_JEF
17.101	QS_M.TS_S.NORM_SUM.TF_F.IDF_F.SIM_JEF
17.101	QS_M.TS_S.NORM_SUM.TF_F.IDF_G.SIM_JEF
17.101	QS_M.TS_S.NORM_SUM.TF_G.IDF_F.SIM_JEF
17.101	QS_M.TS_S.NORM_SUM.TF_G.IDF_G.SIM_JEF

(2.54%). However, we were not able to determine a single variant that clearly outperforms all others. The *IDF* variants most frequently occurring within the top ranks of the music sets are IDF_B2 (13.93%), IDF_J (13.71%), and IDF_E (13.38%). For the movie set C1km, the very same variants perform best (IDF_E with 11.16% occurrence, IDF_J with 11.09%, and IDF_B2 with 9.95%).

As for the similarity measure, we found no clear evidence that cosine similarity (SIM_COS), the de-facto standard measure in IR, generally outperforms the others. It is likely that the key advantage of SIM_COS, the document length normalization, plays a minor role here, because tweets are limited to 140 characters which are usually exhausted by Twitter users. Further support for this hypothesis is given by the remarkably good performance of the simple inner product SIM_INN measure that does not perform any length normalization. On all three data sets, SIM_INN occurs almost twice as often as SIM_COS among the top-ranked variants (about 32 vs. 16%). Also among the virtual document normalization methods, using no normalization at all (NORM_NO) outperforms the other variants investigated, accounting for 52.24% of the top ranks for the music sets, and for 39.94% of the top variants using set C1km. In addition to SIM_INN, also the Jeffrey divergence-based similarity SIM_JEF performed comparably well over all data sets (31.5% for the music sets, 17.77% for C1km).

To investigate if extrapolating the results from the small music set C224a to the real-world set C3ka is valid, we calculated Spearman's rank-order correlation coefficient (e.g., Sheskin 2004) on the two rankings obtained with the two artist sets. The computation revealed a moderate correlation of 0.37. This correlation indicates that the rankings

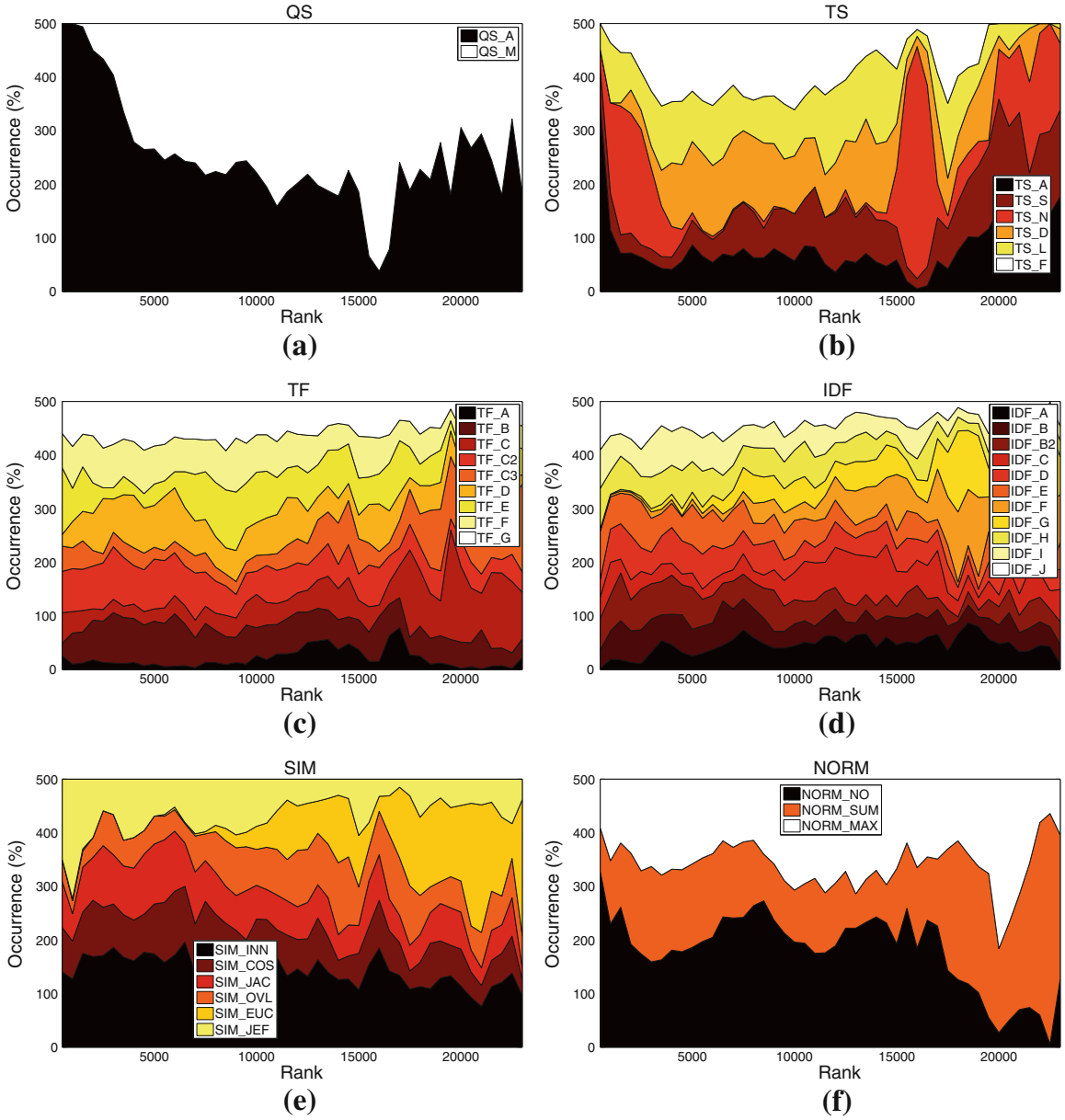


Fig. 4 Distribution of different settings among *all variants* on music set `C224a`. **a** Query scheme, **b** term set, **c** *TF* formulation, **d** *IDF* formulation, **e** similarity function, **f** normalization method

produced by the same algorithmic choices are not largely influenced by factors such as size of artist collection or number of artists per genre.

4.4.3 Average quality and performance variance

In order to assess the quality of individual algorithmic choices—e.g., the use of a specific similarity measure—for the overall task of retrieving similar items, we further computed for all aspects analyzed and for each concrete choice average performance measures over all combinations that use the algorithmic choice under consideration. In particular, *arithmetic mean*, *median*, and *standard deviation* of the *ranks* and the actual *MAP scores* were calculated; mean and median describe the overall performance of each algorithmic choice, whereas the standard deviation can be interpreted as an estimate of the "robustness" of the algorithmic choice against changes in other algorithmic aspects. If

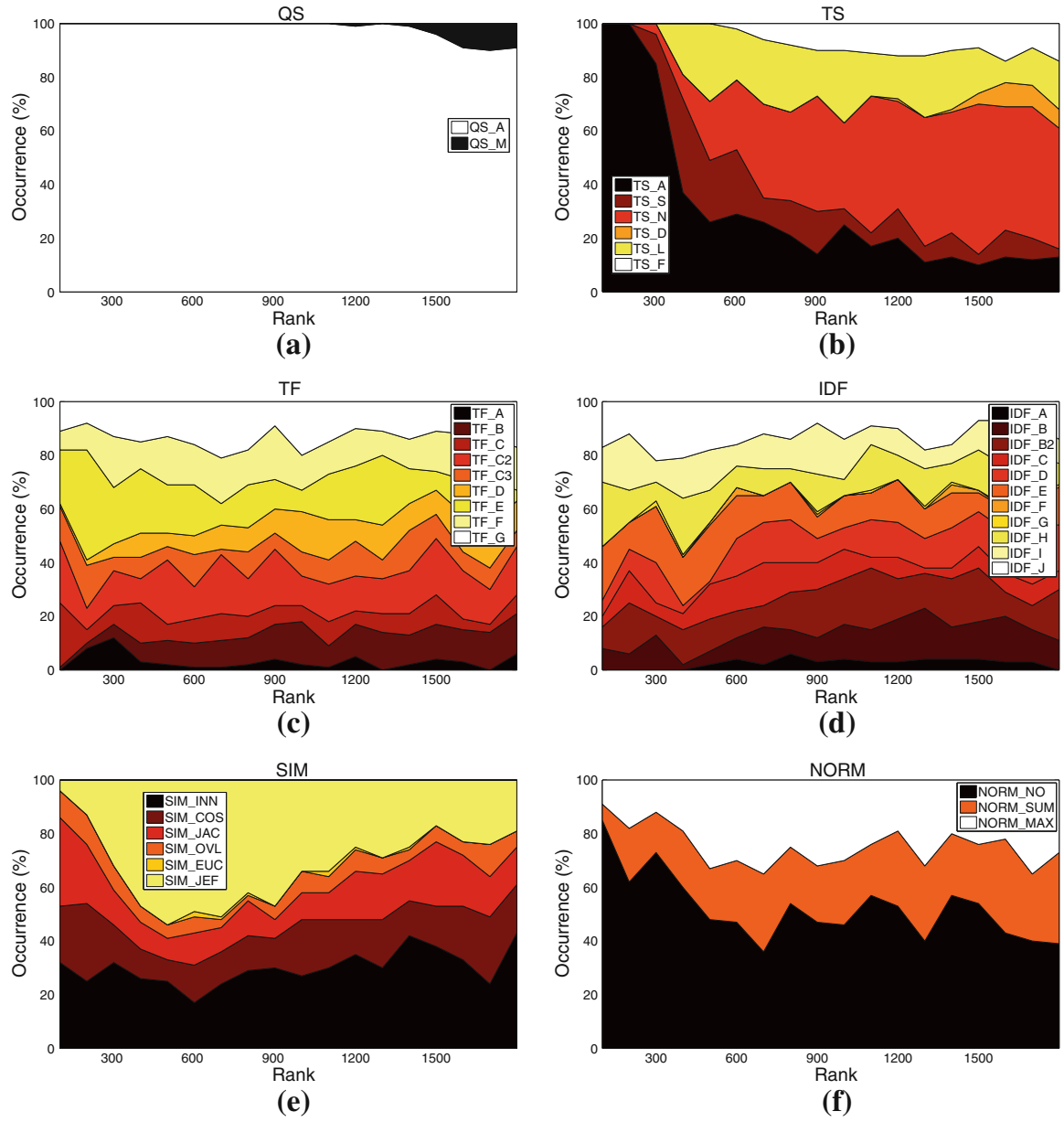


Fig. 5 Distribution of different settings among the *top-ranked variants* on music set `C224a`. **a** Query scheme, **b** term set, **c** *TF* formulation, **d** *IDF* formulation, **e** similarity function, **f** normalization method

variants employing a specific choice are tightly grouped together in the rank-ordered set of all combinations, their standard deviation will be small. This also means that the performance of such tightly grouped variants (according to a particular aspect, e.g., use of term set `TS_F`) is less sensitive to changes in other choices (for example, employing a different normalization).

To investigate both average performance and robustness of specific variants, Fig. 1 shows box plots of the rankings obtained for each algorithmic choice in each of the six broad aspects under consideration.[17] Figure 2 shows the same statistical figures, but this

[17] The red mark represents the median, the upper and lower edges of the box are respectively the 25th and 75th percentiles, the whiskers extend to the most extreme data points not considered outliers (http://www.mathworks.de/help/toolbox/stats/boxplot.html. Accessed December 2011).

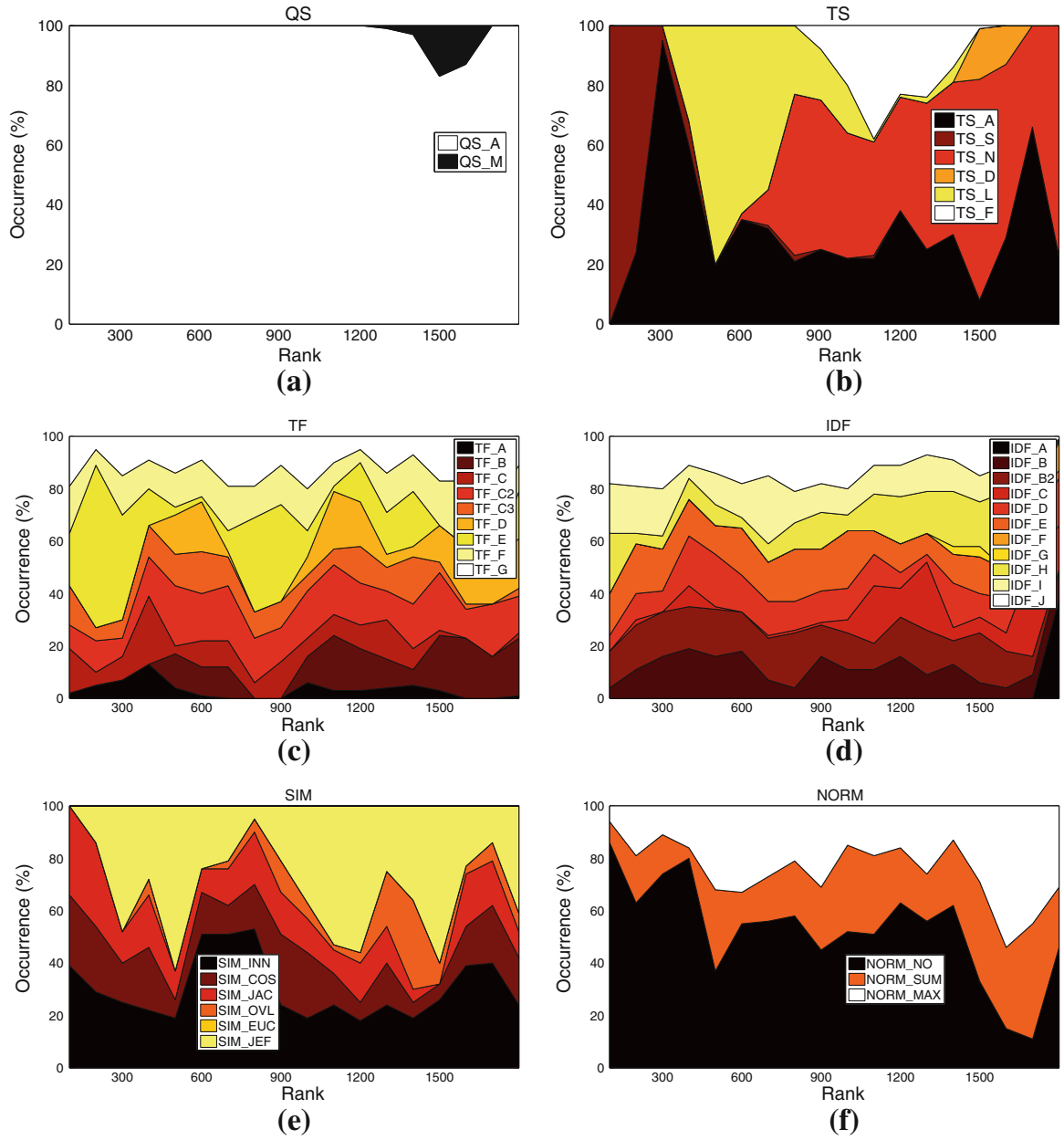


Fig. 6 Distribution of different settings among the *top-ranked variants* on music set `C3ka`. **a** Query scheme, **b** term set, **c** *TF* formulation, **d** *IDF* formulation, **e** similarity function, **f** normalization method

time computed on MAP scores instead of ranks. Table 12 reports detailed results for each algorithmic choice.

Taking a closer look at Figs. 1, 2 and Table 12, the following observations can be made:

- `QS_A` clearly outperforms `QS_M` in terms of quality, although the results obtained with `QS_M` are more robust.
- `TS_F` outperforms all other term sets, both in quality and robustness. This superiority becomes even more clearly visible when using MAP scores as quality measure (Fig. 2) instead of ranks (Fig. 1). Interestingly, term sets `TS_A` and `TS_N` do not perform well overall, since the results they produce are spread across a wide range of ranks (or MAP values), and their quality is not too good either. Figure 4b reveals the reason for the huge spread of `TS_N`: Even though `TS_N` is employed in some of the highest ranked variants, there are also two large clusters of variants employing `TS_N`

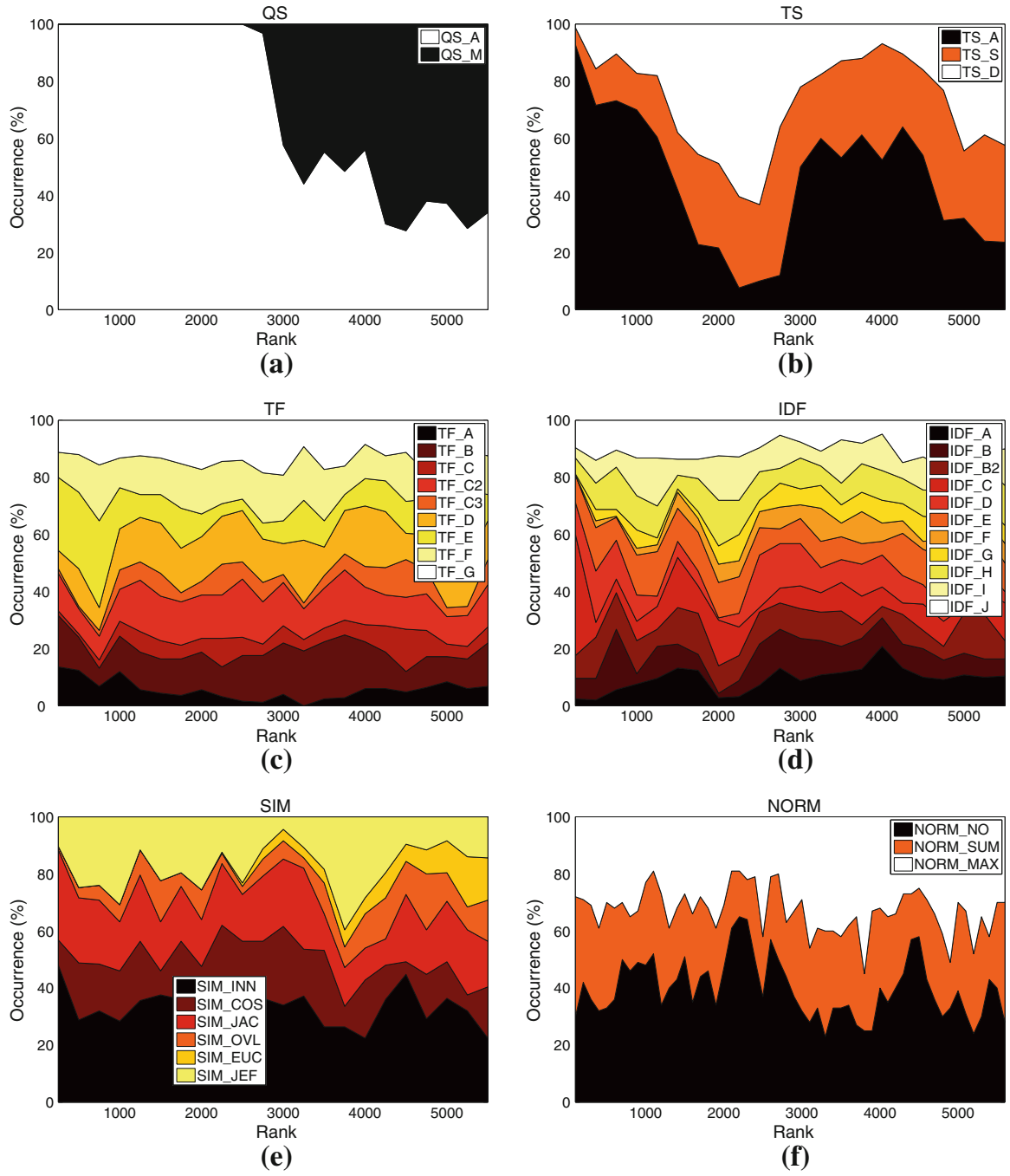


Fig. 7 Distribution of different settings among the *top-ranked variants* on movie set `C1km`. **a** Query scheme, **b** term set, **c** *TF* formulation, **d** *IDF* formulation, **e** similarity function, **f** normalization method

towards the very end of the rank-ordered set of experiments (> rank 15,000). Especially its combination with the algorithmic choices `QS_M`, `TF_B`, `TF_D`, `TF_E`, `IDF_A`, `IDF_H`, `SIM_INN`, or `SIM_OVL` proves detrimental. Looking at the quality scores of `TS_A`, a particularly interesting fact stands out, which is that `TS_A` performs much better in terms of MAP than in terms of rank score. Hence, although the findings presented in Sect. 4.4.2 suggest that `TS_A` is well-suited to yield top results, this seems to be true only when particular other algorithmic choices are present. As a consequence, `TS_A` should be used with caution, only when computational complexity is not an issue and when other algorithmic choices can be ensured (cf. Table 10).

Table 12 Detailed results among algorithmic choices on music set C224a

Variant	Rank					MAP				
	Median	Mean	SD	Min	Max	Median	Mean	SD	Min	Max
QS_A	**8,929**	**10,021**	7,247.2	**1**	**23,080**	**48.066**	**40.380**	18.016	**3.869**	**64.018**
QS_M	13,340	13,080	**5,633.8**	1,112	23,100	39.732	35.399	**15.866**	2.649	57.083
TS_A	12,421	12,257	7,869.8	**1**	23,100	42.158	34.325	20.659	2.649	64.018
TS_D	10,388	10,983	5,349.6	1,112	23,052	45.774	40.693	14.028	4.851	57.083
TS_F	**8,639**	**9,068**	**4,831.1**	529	**19,098**	**48.527**	**45.408**	**10.402**	**12.857**	**58.393**
TS_L	10,343	9,949	5,561.6	318	23050	45.863	43.012	12.610	4.970	59.702
TS_N	15,660	12,880	7,901.3	221	23,058	33.214	33.181	18.853	4.702	60.595
TS_S	14,444	14,165	6,439.0	225	23,098	35.997	30.717	18.527	2.976	60.536
TF_A	13,152	12,310	**5,325.9**	144	**22,979**	40.223	38.829	**12.361**	**5.982**	61.875
TF_B	10,034	10,505	6,373.3	90	23,092	46.399	40.625	15.846	3.452	62.321
TF_C	17,240	14912	6,462.9	3	23,096	26.116	28.360	18.222	3.066	63.839
TF_C2	**9,006**	**9,854**	6,509.2	**1**	23,088	**47.961**	**41.976**	15.709	3.631	**64.018**
TF_C3	17,972	15,421	6,618.3	22	23,093	19.137	26.607	18.628	3.423	63.066
TF_D	9,871	10,371	6,290.5	89	23,053	46.682	40.974	15.697	4.821	62.321
TF_E	10,587	10,981	6,110.3	15	23,063	45.417	40.524	14.978	4.435	63.184
TF_F	9,448	10,079	6,276.5	62	23,100	47.321	41.836	15.187	2.649	62.589
TF_G	9,274	10,028	6,303.5	25	23,097	47.589	41.899	15.288	3.006	62.976
IDF_A	12,855	12,632	5,775.5	449	23,055	41.012	36.061	15.921	4.792	58.839
IDF_B	9,773	10,475	6,681.4	29	23,000	46.860	40.374	16.573	5.804	62.946
IDF_B2	**8,665**	9,780	6,765.0	6	23,058	**48.482**	41.785	16.607	4.702	63.780
IDF_C	12,782	12,409	6,020.9	87	23,027	41.220	37.195	15.609	5.446	62.351
IDF_D	9,790	10,554	6,700.2	13	23,015	46.845	40.208	16.601	5.595	63.363
IDF_E	8,744	9,784	6,750.8	**1**	23,095	48.363	41.722	16.674	3.274	**64.018**
IDF_F	17,108	16,061	**4,884.2**	459	23,100	27.173	26.772	15.586	2.649	58.750
IDF_G	17,009	15,439	5,334.3	258	23,099	28.021	28.219	**16.568**	2.857	60.179
IDF_H	9,731	10,304	6,596.3	19	**22,970**	46.920	40.977	16.472	**6.042**	63.155
IDF_I	8,889	9,887	6,637.9	12	23,038	48.125	41.664	16.435	5.298	63.393
IDF_J	8,673	**9,731**	6,750.9	2	23,096	**48.482**	**41.807**	16.660	3.066	63.929
SIM_COS	9,281	10,275	6,463.8	4	23,060	47.559	41.047	16.073	4.583	63.809
SIM_DIC	9,201	10,127	6,471.4	7	**22,910**	47.679	41.323	16.013	**6.339**	63.780
SIM_EUC	18,116	17,262	**4,082.2**	562	23,027	18.095	23.070	**14.796**	5.446	58.274
SIM_INN	10,896	11,135	6,407.8	141	23,005	44.926	39.429	16.049	5.774	61.905
SIM_JAC	9,194	10,132	6,470.0	**1**	22,972	47.708	41.318	16.017	6.042	**64.018**
SIM_JEF	**8,235**	**9,299**	6,820.6	87	23,058	**49.137**	**42.635**	16.488	4.702	62.351
SIM_OVL	13,004	12,624	6,074.8	18	23,100	40.610	36.403	16.155	2.649	63.155
NORM_MAX	11,851	11,781	6,415.5	27	23,100	43.452	37.418	16.919	2.649	62.976
NORM_NO	**9,482**	**9,811**	**5,908.0**	**1**	**23,054**	**47.292**	**43.276**	**13.289**	**4.821**	**64.018**
NORM_SUM	14,978	13,277	7,219.1	15	23,096	34.241	32.300	19.258	3.066	63.184

Best results for each category are printed in boldface

- As for the term frequency, formulations TF_C and TF_C3 perform poorly and are unstable. We therefore strongly recommend to refrain from these. The binary formulation TF_A is the most stable one, but performs inferior to all but the worst variants mentioned above. Among the other, preferably performing variants, TF_C2 sticks out as yielding particularly good results, in terms of both rank score and MAP. Furthermore, TF_F and TF_G perform equally well as TF_C2 in terms of MAP and slightly worse than the top-performing variant in terms of rank score. Both TF_F and TF_G are slightly more robust than TF_C2. Hence, as an overall recommendation one should select one of the term frequency formulations TF_C2, TF_F, or TF_G, with a slight preference for the former one if top-performance is crucial and a slight preference for one of the latter two variants if stability of the results is more important.
- Variants IDF_A, IDF_C, IDF_F, and IDF_G perform significantly worse than the other formulations of inverse document frequency. As for top-performing choices, IDF_E ranks at the very top according to both MAP and rank scores. Also IDF_B2 and IDF_J are not significantly inferior.
- Among the similarity functions, SIM_EUC performs remarkably inferior to all other variants. SIM_OVL does not perform considerably better. Best results can be achieved employing SIM_JEF, while at the same time maintaining a reasonable stability level.
- NORM_NO performs best in terms of quality and robustness, whereas NORM_SUM performs worst in both regards.

4.4.4 Comparison with web page-based experiments

We also conducted a similar study using as data source Web pages related to music artists instead of microblogs (Schedl et al. 2011). In order to assess the specificities of microblogs, in the following the results obtained in the paper at hand for the music data sets are compared against those reported in Schedl et al. (2011), where the same evaluation setting is employed. Although the music data sets are partly different, the results of Schedl et al. (2011) are comparable to those of the current study. Overall, the best-performing variants according to Schedl et al. (2011) in this paper's notation are the following:

- TF_C3.IDF_I.SIM_COS
- TF_C3.IDF_H.SIM_COS
- TF_C2.IDF_I.SIM_COS
- TF_C2.IDF_H.SIM_COS

In all top-ranked variants, no normalization on the Web page-level, i.e., giving each Web page retrieved for the artist under consideration the same weight, is performed. Nevertheless, the virtual documents are normalized, i.e., when aggregating individual Web pages retrieved for a particular artist to a virtual document, each term score is divided by the absolute number of Web pages retrieved for the artist that contain the term.

Comparing the two studies, the first observation to be made is that regardless of the data source (Web pages or microblogs), logarithmic formulations of *TF* tend to perform best (in particular for music artists). As for *IDF*, the variants IDF_I, IDF_H, and IDF_B2 perform best for Web pages, while IDF_B2, IDF_E, and IDF_J yield highest MAP scores for microblogs. Thus, again logarithmic formulations considerably outperform other variants for both data sources. Regarding the similarity measure, the top-ranked variants on the corpus of Web pages employ cosine similarity, while for microblogs no clear indication

for the cosine measure to outperform the others can be found. Furthermore, normalization does not improve results when the corpus is constituted of tweets. In contrast, when the corpus comprises Web pages, normalization on the level of virtual documents considerably ameliorates the MAP scores. No comparison can be made on the level of term sets due to the fact that Schedl et al. (2011) does not take into account different dictionaries for indexing. As for the query scheme, `QS_M` which includes the term "music" in addition to the named entity sought for clearly outperforms `QS_A` on the Web-page-corpus, while the inverse holds on the microblog-corpus. It seems that adding additional, domain-specific search terms to the query is counterproductive when looking for microblogs since it prunes the set of tweets too heavily, while doing so is a necessity to filter unrelated Web pages from the search results.

4.5 Alternative classifiers

Since we modeled and evaluated the retrieval task as a genre classification task, we can alternatively use classifiers other than kNN for evaluation purposes. We hence compare the memory-based kNN classifier with several state-of-the-art classifiers: the kernel-based *Support Vector Machines* (SVM) (Vapnik 1995), *Random Forests* (RF) (Breiman 2001), i.e., an ensemble learner based on decision trees, and *Repeated Incremental Pruning to Produce Error Reduction* (RIPPER) (Cohen 1995), a propositional rule learner. We

Table 13 Accuracies of the top-ranked and bottom-ranked variants using SVM classification on music set `C224a`

Acc.	Variant
71.875	`QS_A.TS_A.NORM_NO.TF_C.IDF_G`
71.875	`QS_A.TS_A.NORM_NO.TF_C2.IDF_G`
71.429	`QS_A.TS_A.NORM_NO.TF_C3.IDF_G`
70.089	`QS_A.TS_A.NORM_NO.TF_C2.IDF_F`
69.643	`QS_A.TS_A.NORM_MAX.TF_F.IDF_G`
69.196	`QS_A.TS_A.NORM_NO.TF_C.IDF_F`
68.750	`QS_A.TS_A.NORM_MAX.TF_B.IDF_F`
68.750	`QS_A.TS_A.NORM_MAX.TF_D.IDF_F`
68.750	`S_A.TS_A.NORM_MAX.TF_G.IDF_F`
68.750	`QS_A.TS_A.NORM_MAX.TF_G.IDF_G`
68.750	`QS_A.TS_A.NORM_NO.TF_D.IDF_F`
68.750	`QS_A.TS_A.NORM_SUM.TF_D.IDF_F`
...	...
7.143	`QS_A.TS_S.NORM_MAX.TF_A.IDF_F`
7.143	`QS_A.TS_S.NORM_MAX.TF_E.IDF_F`
7.143	`QS_A.TS_S.NORM_NO.TF_A.IDF_F`
7.143	`QS_A.TS_S.NORM_NO.TF_E.IDF_F`
7.143	`QS_A.TS_S.NORM_NO.TF_F.IDF_F`
7.143	`QS_A.TS_S.NORM_NO.TF_G.IDF_F`
7.143	`QS_A.TS_S.NORM_SUM.TF_A.IDF_F`
7.143	`QS_A.TS_S.NORM_SUM.TF_E.IDF_F`
6.696	`QS_A.TS_S.NORM_MAX.TF_C.IDF_F`
6.696	`QS_A.TS_S.NORM_MAX.TF_C3.IDF_F`

Table 14 Accuracies of the top-ranked and bottom-ranked variants using RF classification on music set C224a

Acc.	Variant
57.589	QS_A.TS_N.NORM_MAX.TF_A.IDF_F
57.589	QS_A.TS_N.NORM_SUM.TF_E.IDF_F
57.589	QS_A.TS_N.NORM_NO.TF_A.IDF_F
57.589	QS_A.TS_N.NORM_SUM.TF_A.IDF_F
57.589	QS_A.TS_N.NORM_MAX.TF_E.IDF_F
57.589	QS_A.TS_N.NORM_NO.TF_E.IDF_F
57.143	QS_A.TS_N.NORM_SUM.TF_G.IDF_F
56.696	QS_A.TS_N.NORM_NO.TF_F.IDF_F
55.357	QS_A.TS_N.NORM_MAX.TF_C2.IDF_F
54.911	QS_A.TS_N.NORM_NO.TF_A.IDF_G
54.911	QS_A.TS_N.NORM_NO.TF_A.IDF_G
54.911	QS_A.TS_N.NORM_SUM.TF_A.IDF_G
54.911	QS_A.TS_N.NORM_MAX.TF_A.IDF_G
54.911	QS_A.TS_N.NORM_MAX.TF_C2.IDF_G
54.911	QS_A.TS_N.NORM_SUM.TF_C2.IDF_G
...	...
7.589	QS_A.TS_S.NORM_MAX.TF_C2.IDF_F
7.143	QS_A.TS_S.NORM_SUM.TF_A.IDF_F
7.143	QS_A.TS_S.NORM_MAX.TF_A.IDF_F
7.143	QS_A.TS_S.NORM_MAX.TF_F.IDF_F
7.143	QS_A.TS_S.NORM_NO.TF_B.IDF_F
7.143	QS_A.TS_S.NORM_SUM.TF_C2.IDF_F
7.143	QS_A.TS_S.NORM_NO.TF_A.IDF_F
7.143	QS_A.TS_S.NORM_NO.TF_C2.IDF_F
6.697	QS_A.TS_S.NORM_SUM.TF_G.IDF_F
6.697	QS_A.TS_S.NORM_SUM.TF_B.IDF_F
6.250	QS_A.TS_S.NORM_MAX.TF_C.IDF_F

employed 10-fold Cross-Validation using the default parameters of the respective WEKA (Hall et al. 2009) classifiers.

Tables 13, 14, and 15 show the highest- and lowest-ranked variants when using as classifier SVM, RF, and RIPPER, respectively. Similar to the kNN experiments described in Sect. 4.3, query set QS_A clearly outperforms QS_M. It can be observed that SVM benefits from having access to as much data as possible, i.e., it achieves highest accuracies when operating on term set TS_A. The Random Forest classifier yields significantly lower accuracies and performs best when using artist names as term set TS_N. The rule learner RIPPER seemingly performs best on the Freebase set TS_F, the reason for which is probably the clearest semantic distinction between the terms in this dictionary. Performing no normalization proved beneficial also for classifiers other than kNN, although in the case of RF, this becomes apparent better from looking at the top ranks in Fig. 9e than from Table 14. To yield top performance with the RF classifier, the use of IDF_F (in addition to QS_A.TS_N) seems to be more important than employing a particular normalization function. No clear picture emerges, in contrast, when analyzing the impact of the term frequency formulation. Even though the top 4 performers with SVM employ variants of the TF_C formulation, combinations including several

Table 15 Accuracies of the top-ranked and bottom-ranked variants using RIPPER classification on music set C224a

Acc.	Variant
58.4821	QS_A.TS_F.NORM_NO.TF_B.IDF_H
58.4821	QS_A.TS_F.NORM_NO.TF_C3.IDF_H
58.0357	QS_A.TS_F.NORM_MAX.TF_C.IDF_C
58.0357	QS_A.TS_F.NORM_MAX.TF_C.IDF_D
58.0357	QS_A.TS_F.NORM_MAX.TF_C.IDF_E
58.0357	QS_A.TS_F.NORM_MAX.TF_C.IDF_J
57.5893	QS_A.TS_F.NORM_NO.TF_C.IDF_J
57.5893	QS_A.TS_F.NORM_NO.TF_C2.IDF_I
57.5893	QS_A.TS_F.NORM_NO.TF_C.IDF_A
57.5893	QS_A.TS_F.NORM_NO.TF_C2.IDF_A
57.5893	QS_A.TS_F.NORM_NO.TF_C.IDF_B2
57.5893	QS_A.TS_F.NORM_NO.TF_B.IDF_D
57.5893	QS_A.TS_F.NORM_NO.TF_C2.IDF_D
57.5893	QS_A.TS_F.NORM_NO.TF_B.IDF_C
57.5893	QS_A.TS_F.NORM_NO.TF_B.IDF_A
57.5893	QS_A.TS_F.NORM_NO.TF_C3.IDF_B
...	...
4.4643	QS_A.TS_S.NORM_SUM.TF_G.IDF_F
4.4643	QS_A.TS_S.NORM_NO.TF_E.IDF_F
4.4643	QS_A.TS_S.NORM_NO.TF_B.IDF_F
4.4643	QS_A.TS_S.NORM_SUM.TF_C.IDF_F
4.4643	QS_A.TS_D.NORM_MAX.TF_A.IDF_G
4.4643	QS_A.TS_S.NORM_NO.TF_G.IDF_F
4.4643	QS_A.TS_S.NORM_SUM.TF_B.IDF_F
4.4643	QS_A.TS_S.NORM_NO.TF_F.IDF_F
4.4643	QS_A.TS_S.NORM_SUM.TF_E.IDF_F
4.4643	QS_A.TS_S.NORM_SUM.TF_C2.IDF_F
4.4643	QS_A.TS_D.NORM_SUM.TF_A.IDF_G
4.4643	QS_A.TS_S.NORM_MAX.TF_G.IDF_F
4.4643	QS_A.TS_S.NORM_MAX.TF_C2.IDF_F
4.4643	QS_A.TS_S.NORM_NO.TF_A.IDF_F
4.4643	QS_A.TS_S.NORM_NO.TF_C2.IDF_F
4.4643	QS_A.TS_S.NORM_SUM.TF_D.IDF_F

other formulations can be found among the top-performing variants as well. Among the top-performing variants in the RF experiments, the simple binary match function TF_A appears surprisingly often. For the decision tree learner, the *IDF* formulation hence seems to be more important. For RIPPER, variants of TF_C clearly outperform all other choices.

Figures 8, 9, and 10 show the distribution of each algorithmic choice among all 3,564 experimental setups[18] when using classifier SVM, RF, and RIPPER, respectively.

[18] For the experiments with alternative classifiers, similarity functions did not apply; if a classifier required a similarity function, we used WEKA's default.

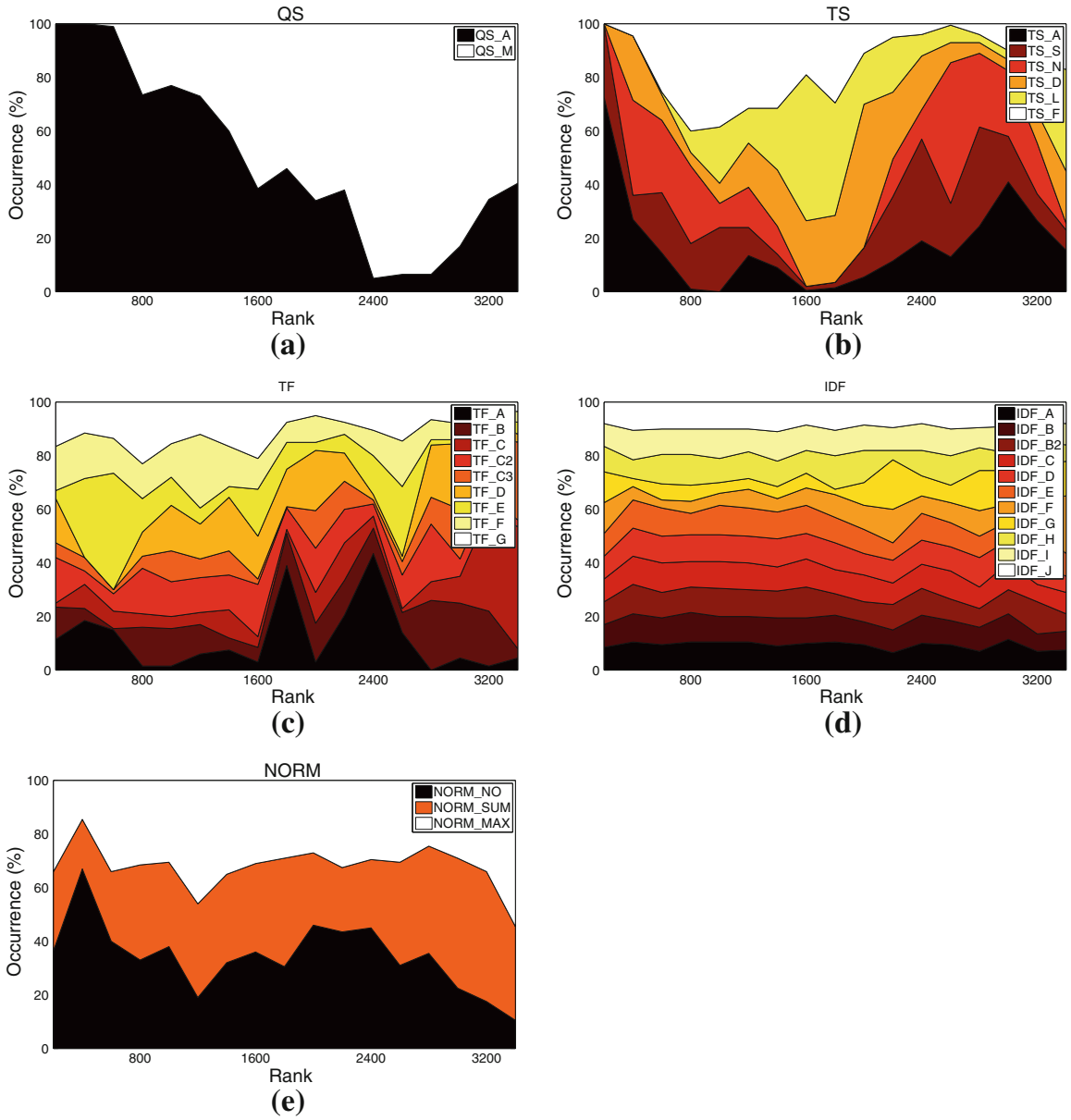


Fig. 8 Distribution of different settings among *all variants* using SVM classification on music set C224a. **a** Query scheme, **b** term set, **c** *TF* formulation, **d** *IDF* formulation, **e** normalization method

Although these plots do not reveal significant information for all aspects analyzed, we can summarize the interesting observations and consequently formulate advices as follow:

- QS_A clearly outperforms QS_M with all classifiers.
- TS_A is found frequently among the top ranks in the SVM experiments, but also peaks at the very bottom ranks. The top 600 ranks of the RF experiments are entirely dominated by TS_N, and TS_F performs very well with the RIPPER classifier. The generic but broad vocabularies TS_A and TS_S perform remarkably inferior when using RF or RIPPER. It seems that rule learners and decision tree learners benefit from a smaller, but more well defined vocabulary, such as TS_N or TS_F.
- It is hard to give advice for favoring or refraining from specific choices of the term frequency function. When using an SVM classifier, Fig. 8c might suggest to employ

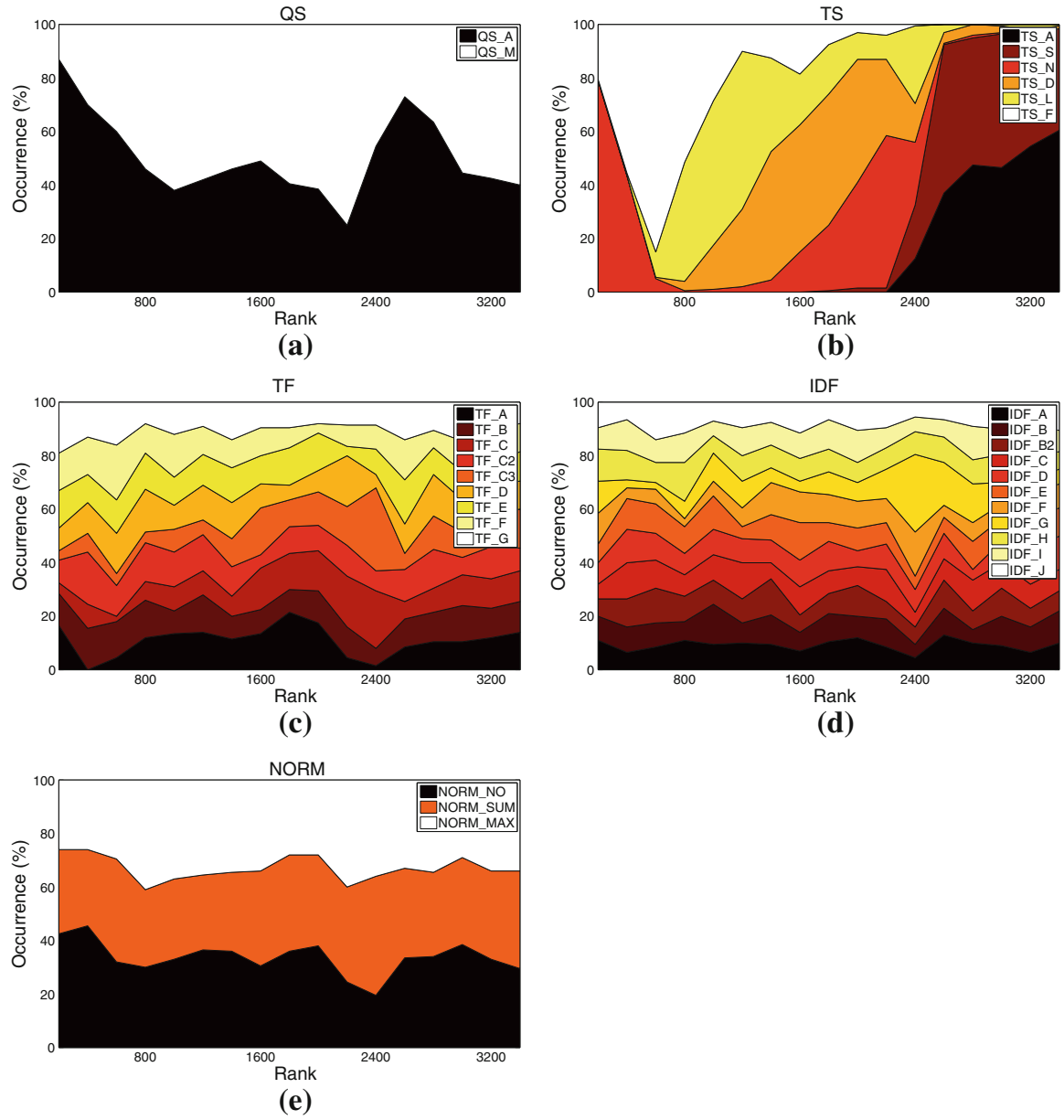


Fig. 9 Distribution of different settings among *all variants* using RF classification on music set C224a. **a** Query scheme, **b** term set, **c** *TF* formulation, **d** *IDF* formulation, **e** normalization method

TF_E or TF_G, because both are frequently found among the top-ranked variants; however, neither of them does consistently perform well. In particular TF_E also occupies inferior positions around rank 2,600. For the RF classifier, TF_G seems the most favorable *TF* formulation, too. One clear advice that can be given is to refrain from TF_A, regardless of the classifier applied. Even though binary match performs well in some settings, the peaks at mediocre and lowest ranks do by no means suggest the use of TF_A.

– The rather uniform distribution of the *IDF* variants among all ranks does not encourage the formulation of specific advices.
– Slight (when using RF or RIPPER) to rather dominant (SVM) peaks of the NORM_NO setting, correspond well to the observation already made for the kNN experiments

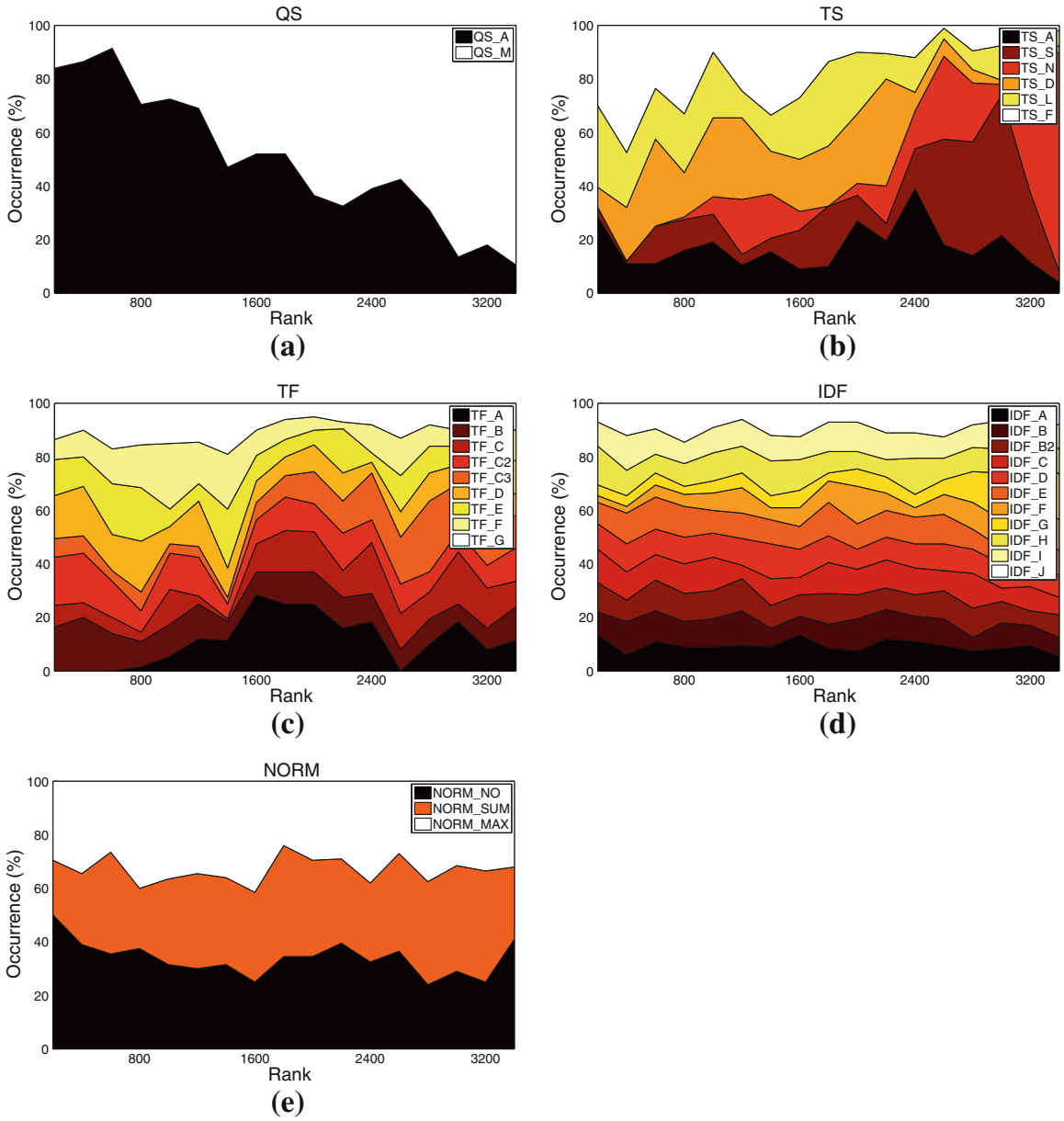


Fig. 10 Distribution of different settings among *all variants* using RIPPER classification on music set `C224a`. **a** Query scheme, **b** term set, **c** *TF* formulation, **d** *IDF* formulation, **e** normalization method

using MAP as performance measure. Due to the special characteristics of tweets, it is not advisable to perform document length normalization.

To investigate whether results are consistent between different classifiers in terms of the variants' rank-order according to classification accuracy, we computed Spearman's rank-order correlation coefficient. The pairwise correlation can be found in Table 16. As it can be seen, different classifiers not very surprisingly yield different ranks for the same algorithmic variants. Nevertheless, a small but significant ($p = 0.00002$) correlation between SVM and RF could be observed. A moderate to high correlation between SVM and RIPPER is notable as well. Between RF and RIPPER a slight to moderate correlation is present. The p values for the combinations (SVM,RIPPER) and (RF,RIPPER) are infinitesimally small.

Table 16 Pairwise Spearman's rank-order correlation coefficients between variants produced by alternative classifiers on music set C224a

	SVM	RF	RIPPER
SVM		0.071	0.528
RF	0.071		0.189
RIPPER	0.528	0.189	

5 Conclusions and future work

In this article, we presented a comprehensive evaluation of using Twitter posts for the purpose of similarity estimation between named entities. To this end, we performed tens of thousands single experiments on three data sets, two related to the music domain, one from the movie domain. Different algorithmic choices related to query scheme, index term set, length normalization, *TF* · *IDF* formulation, and similarity measure were thoroughly investigated. The main findings can be summarized as follows:

- Restricting the search by domain-specific key words prunes the resulting set of tweets too heavily. Using only the named entity as query (QS_A) should be favored.
- Top-ranked results are achieved using all terms in the corpus (TS_A), though at high computational costs and little robustness against small changes in other algorithmic choices. If computational complexity or robustness is an issue, the results suggest using as index term set a domain-specific dictionary (TS_F for the music domain or TS_D for the movie domain).
- Normalizing for length does not significantly improve the results, neither when performed on term vectors, nor when included in the similarity function. Taking into account the higher computational costs, we therefore recommend refraining from normalization (NORM_NO) and using as similarity measure, for example, the inner product (SIM_INN) or the Jeffrey divergence-based similarity (SIM_JEF). Both SIM_EUC and SIM_OVL should definitively be avoided.
- The binary match *TF* formulation TF_A should not be used. The most favorable variants are TF_C2 and TF_E. But also TF_F and TF_G do not perform significantly worse, regardless of the data set used.
- Among the *IDF* formulations, we suggest to refrain from using IDF_A, IDF_F, and IDF_G, as they performed poorly on all data sets. Better alternatives are given by formulations IDF_B2, IDF_E, and IDF_J, which ranked well on all sets.

Future work on evaluating different similarity models based on microblogs will include incorporating the bloger's perspective, for example, by exploiting social graphs. Taking into account that perceived similarities are often subjective, influenced by peers, and can be defined according to very different dimensions, in the music as well as in the movie domain, a more fine-grained analysis based on the results presented here should be performed. As some of the algorithmic choices of the best- and worst-performing combinations varied between the movie and music data sets, we further plan to assess if the performance of specific variants depends on the type of the named entities. We will therefore conduct experiments on other sets of named entities, for example, politicians or books.

Another promising research direction is assessing *temporal and geographic properties of tweets*. Geographical aspects could be used, for example, to develop *geo-aware popularity estimates* of named entities. Together with temporal information, such a popularity

measure could give indication on the development and spreading of trends around the world.

Acknowledgments The author would like to express his gratitude to Tim Pohle for providing parts of the source code for evaluation. He further wishes to acknowledge the reviewers for their profound and very helpful comments. This research is supported by the Austrian Science Funds (FWF): P22856-N23.

References

Armentano, M. G., Godoy, D., & Amandi, A. A. (2011). Recommending information sources to information seekers in Twitter. In *Proceedings of the IJCAI 2011: International workshop on social web mining*, Barcelona, Spain.

Aucouturier, J. J., & Pachet, F. (2002). Scaling up music playlist generation. In *Proceedings of the IEEE international conference on multimedia and expo (ICME)*.

Baeza-Yates, R., & Ribeiro-Neto, B. (2011). *Modern information retrieval—the concepts and technology behind search* (2nd ed.). Harlow, UK: Addison Wesley.

Baumann, S., & Hummel, O. (2003). Using cultural metadata for artist recommendation. In *Proceedings of the 3rd international conference on web delivering of music (WEDELMUSIC)*.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Machine Learning Research, 3*, 993–1022.

Breiman, L. (2001). Random forests. *Machine Learning, 45*, 5–32.

Brill, E. (1992). A simple rule-based part of speech tagger. In *Proceedigs of the 3rd conference on applied natural language processing* (pp. 152–155).

Buckley, C., & Voorhees, E. (2000). Evaluating evaluation measure stability. In *Proceedings of the 23rd annual international ACM SIGIR conference on research and development in information retrieval (SIGIR)*.

Celma, O. (2008). *Music recommendation and discovery in the long tail*. Ph.D. thesis, Universitat Pompeu Fabra, Barcelona, Spain.

Celma, O., Cano, P., & Herrera, P. (2006). Search sounds: An audio crawler focused on weblogs. In *Proceedings of the 7th international conference on music information retrieval (ISMIR)*, Victoria, Canada.

Cheng, Z., Caverlee, J., & Lee, K. (2010). You are where you tweet: A content-based approach to geo-locating Twitter users. In *Proceedings of the 19th ACM international conference on information and knowledge management (CIKM)*.

Cimiano, P., Handschuh, S., & Staab, S. (2004). Towards the self-annotating web. In *Proceedings of the 13th international conference on World Wide Web (WWW)*.

Cimiano, P., & Staab, S. (2004). Learning by googling. *ACM SIGKDD Explorations Newsletter, 6*(2), 24–33.

Cohen, W. W. (1995). Fast and effective rule induction. In *Proceedings of the 12th international conference on machine learning (ICML)*, Lake Tahoe, CA.

Cohen, W. W., & Fan, W. (2000). Web-collaborative filtering: Recommending music by crawling the web. *WWW9/Computer Networks, 33*, 1–6.

Debole, F., & Sebastiani, F. (2003). Supervised term weighting for automated text categorization. In *Proceedings the 18th ACM symposium on applied computing (SAC)*.

Dong, A., Zhang, R., Kolari, P., Bai, J., Diaz, F., Chang, Y., et al. (2010). Time is of the essence: Improving recency ranking using Twitter data. In *Proceedings of the 19th international conference on World Wide Web (WWW)* (pp. 331–340), Raleigh, NC.

Duan, Y., Jiang, L., Qin, T., Zhou, M., & Shum, H. (2010). An empirical study on learning to rank of tweets. In *Proceedings of the 23rd international conference on computational linguistics (COLING)* (pp. 295–303), Beijing, China.

Evans, M. (2011). Twitter enjoys major growth and excellent stickiness. http://blog.sysomos.com/2010/03/29/twitter-enjoys-major-growth-and-excellent-stickiness. Accessed January 2011.

Geleijnse, G., & Korst, J. (2006). Web-based artist categorization. In *Proceedings of the 7th international conference on music information retrieval (ISMIR)*.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten I. H. (2009). The WEKA data mining software: An update. *SIGKDD Explorations Newsletter, 11*, 10–18.

Hu, X., Downie, J. S., West, K., & Ehmann, A. (2005). Mining music reviews: Promising preliminary results. In *Proceedings of the 6th international conference on music information retrieval (ISMIR)*, London, UK.

Hu, X., & Downie, J. S. (2007). Exploring mood metadata: Relationships with genre, artist and usage metadata. In *Proceedings of the 8th international conference on music information retrieval (ISMIR)*.

Jass, G. (2003). http://www.imdb.com. Accessed January 2011.

Java, A., Song, X., Finin, T., & Tseng, B. (2007). Why we Twitter: Understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD workshop on web mining and social network analysis*.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM, 46*(5), 604–632.

Knees, P., Pampalk, E., & Widmer, G. (2004). Artist classification with web-based data. In *Proceedings of the 5th international symposium on music information retrieval (ISMIR)*.

Knees, P., Pohle, T., Schedl, M., & Widmer, G. (2007). A music search engine built upon audio-based and web-based similarity measures. In *Proceedings of 30th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR)*.

Knees, P., Schedl, M., Pohle, T., & Widmer, G. (2007). Exploring music collections in virtual landscapes. *IEEE MultiMedia, 14*(3), 46–54.

Knees, P., Schedl, M., & Pohle, T. (2008). A deeper look into web-based classification of music artists. In *Proceedings of 2nd workshop on learning the semantics of audio signals (LSAS)*.

Koren, Y. (2009). Collaborative filtering with temporal dynamics. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining (KDD)* (pp. 447–456), Paris, France.

Kwak, H., Lee, C., Park, H., & Moon, S. (2010). What is Twitter, a social network or a news media? In *Proceedings of the 19th international conference on World Wide Web (WWW)*.

Lan, M., Tan, C. L., Low, H. B., & Sung, S. Y. (2005). A comprehensive comparative study on term weighting schemes for text categorization with support vector machines. In *Special interest tracks and posters of the 14th international conference on World Wide Web (WWW)*.

Lin, J. (1991). Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory, 37*, 145–151.

Luhn, H. P. (1957). A statistical approach to mechanized encoding and searching of literary information. *IBM Journal, 1*(4), 309–317.

Pachet, F., & Cazaly, D. (2000). A taxonomy of musical genre. In *Proceedings of content-based multimedia information access (RIAO) conference*.

Pampalk, E., Flexer, A., & 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)*.

Pampalk, E., & Goto, M. (2007). MusicSun: A new approach to artist recommendation. In*Proceedings of the 8th international conference on music information retrieval (ISMIR)*.

Pérez-Iglesias, J., Pérez-Agüera, J. R., Fresno, V., & Feinstein, Y. Z. (2009). Integrating the probabilistic models BM25/BM25F into Lucene. CoRR abs/0911.5046.

Pohle, T., Knees, P., Schedl, M., Pampalk, E., & Widmer, G. (2007). "Reinventing the wheel": A novel approach to music player interfaces. *IEEE Transactions on Multimedia, 9*, 567–575.

Robertson, S., Walker, S., & Beaulieu, M. (1999). Okapi at TREC-7: Automatic ad hoc, filtering, VLC and interactive track. In *Proceedings of the 7th text retrieval conference (TREC-7)*.

Robertson, S., Walker, S., & Hancock-Beaulieu, M. (1995). Large test collection experiments on an operational, interactive system: Okapi at TREC. *Information Processing & Management, 31*(3), 345–360.

Sakaki, T., Okazaki, M., & Matsuo, Y. (2010). Earthquake shakes Twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th international conference on World Wide Web (WWW)*.

Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM, 18*(11), 613–620.

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing and Management, 24*(5), 513–523.

Sanderson, M., & Zobel, J. (2005). Information retrieval system evaluation: Effort, sensitivity, and reliability. In *Proceedings of 28th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR)*.

Sankaranarayanan, J., Samet, H., Teitler, B. E., Lieberman, M. D., & Sperling, J. (2009). Twitterstand: News in tweets. In *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems (GIS)* (pp. 42–51), Seattle, WA.

Schedl, M. (2008). *Automatically extracting, analyzing, and visualizing information on music artists from the World Wide Web*. Ph.D. thesis, Johannes Kepler University Linz, Linz, Austria.

Schedl, M. (2010). On the use of microblogging posts for similarity estimation and artist labeling. In *Proceedings of the 11th international society for music information retrieval conference (ISMIR)*, Utrecht, the Netherlands.

Schedl, M. (2011). Analyzing the potential of microblogs for spatio-temporal popularity estimation of music artists. In *Proceedings of the IJCAI 2011: International workshop on social web mining*, Barcelona, Spain.

Schedl, M., & Knees, P. (2008). Investigating different term weighting functions for browsing artist-related web pages by means of term co-occurrences. In *Proceedings of the 2nd international workshop on learning the semantics of audio signals (LSAS)*.

Schedl, M., Knees, P., Seyerlehner, K., & Pohle, T. (2007). The CoMIRVA toolkit for visualizing music-related data. In *Proceedings of the 9th Eurographics/IEEE VGTC symposium on visualization (EuroVis)*.

Schedl, M., Knees, P., & Widmer, G. (2005). A web-based approach to assessing artist similarity using co-occurrences. In *Proceedings of the 4th international workshop on content-based multimedia indexing (CBMI)*.

Schedl, M., Pohle, T., Knees, P., & Widmer, G. (2011). Exploring the music similarity space on the web. *ACM Transactions on Information Systems, 29*(3), 14:1–14:24.

Schedl, M., Widmer, G., Knees, P., & Pohle, T. (2011). A music information system automatically generated via web content mining techniques. *Information Processing & Management, 47*, 426–439.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys, 34*(1), 1–47.

Sharifi, B., Hutton, M. A., & Kalita, J. (2010). Summarizing microblogs automatically. In *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics (NAACL HLT)*.

Sheskin, D. J. (2004). *Handbook of parametric and nonparametric statistical procedures* (3rd ed.). Boca Raton, London, New York, Washington, DC: Chapman & Hall/CRC.

Sriram, B., Fuhry, D., Demir, E., Ferhatosmanoglu, H., & Demirbas, M. (2010). Short text classification in twitter to improve information filtering. In *Proceedings of the 33th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR)*, Geneva, Switzerland.

Teevan, J., Ramage, D., & Morris, M. R. (2011). #TwitterSearch: A comparison of microblog search and web search. In *Proceedings of the 4th ACM international conference on web search and data mining (WSDM)*, Hong Kong, China.

Turnbull, D., et al. (2007). Towards musical query-by-semantic description using the CAL500 data set. In *Proceedings of 30th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR)*.

Vapnik, V. N. (1995). *The nature of statistical learning theory*. Berlin: Springer.

Weng, J., Lim, E. P., Jiang, J., & He, Q. (2010). TwitterRank: Finding topic-sensitive influential twitterers. In *Proceedings of the 3th ACM international conference on web search and data mining (WSDM)*, New York, NY.

Whitman, B., & Lawrence, S. (2002). Inferring descriptions and similarity for music from community metadata. In *Proceedings of the international computer music conference (ICMC)*.

Yarow, J. (2011). *Twitter finally reveals all its secret stats*. http://www.businessinsider.com/twitter-stats-2010-4. Accessed January 2011.

Zadel, M., & Fujinaga, I. (2004). Web services for music information retrieval. In *Proceedings of the 5th international symposium on music information retrieval (ISMIR)*, Barcelona, Spain.

Zobel, J., & Moffat, A. (1998). Exploring the similarity space. *ACM SIGIR Forum, 32*(1), 18–34.

Zobel, J., & Moffat, A. (2006). Inverted files for text search engines. *ACM Computing Surveys, 38*(2), 1–55.

Markus Schedl, Christian Höglinger, Peter Knees

Large-Scale Music Exploration in Hierarchically Organized Landscapes Using Prototypicality Information

Proceedings of the ACM International Conference on Multimedia Retrieval (ICMR)

Trento, Italy, April 2011

Large-Scale Music Exploration in Hierarchically Organized Landscapes Using Prototypicality Information

Markus Schedl, Christian Höglinger, Peter Knees

Department of Computational Perception
Johannes Kepler University
Linz, Austria
markus.schedl@jku.at

ABSTRACT

We present a novel user interface that offers a fun way to explore music collections in virtual landscapes in a game-like manner. Extending previous work, special attention is paid to scalability and user interaction. In this vein, the ever growing size of today's music collections is addressed in two ways that allow for visualizing and browsing nearly arbitrarily sized music repositories. First, the proposed user interface `deepTune` employs a hierarchical version of the Self-Organizing Map (SOM) to cluster similar pieces of music using multiple, hierarchically aligned layers. Second, to facilitate orientation in the landscape by presenting well-known anchor points to the user, a combination of Web-based and audio signal-based information extraction techniques to determine cluster prototypes (songs) is proposed. Selecting representative and well-known prototypes – the former is ensured by using signal-based features, the latter by using Web-based data – is crucial for browsing large music collections. We further report on results of an evaluation carried out to assess the quality of the proposed cluster prototype ranking.

Categories and Subject Descriptors: H.5.2 [User Interfaces]: Auditory, Graphical user interfaces H.5.1 [Multimedia Information Systems]: Artificial, augmented, and virtual realities

General Terms: Algorithms

Keywords: music information extraction, user interface, human–computer interaction, unsupervised learning

1. BACKGROUND AND RELATED WORK

Steadily growing sizes of digital music collections, both in the private and the commercial area, necessitate intelligent user interfaces [18] to make the vast amount of music available to all. Methods for exploring music repositories by means beyond simple text-based interfaces[1] are thus gaining more and more popularity. According to [33], music retrieval systems to access music collections can be broadly categorized with respect to the employed *query formulation* method into *direct querying*, *query by example*, and *browsing* systems. The approach presented in the paper at hand uses the modality of browsing to retrieve music from a possibly huge repository.

One group of algorithms that aims at offering the user a means of intelligently exploring music collections is *music recommender systems*, e.g., [4]. Such a system is usually built by first deriving features from various sources, such as tags obtained via game playing [9, 32], artist term profiles extracted from Web pages [2, 12], or RSS feeds [5]. Subsequently, a similarity measure between artists or between songs is applied to the feature vectors. The resulting similarity estimates are then used to recommend music similar to a given input song or artist. As an alternative or in addition, collaborative filtering techniques [3] may be used for or incorporated into the recommendation.

1.1 Intelligent User Interfaces to Music

Another category of approaches to transcend traditional, text-based ways of music retrieval is *intelligent user interfaces* (IUI) [18] to explore music collections. The `deepTune` application presented here is one example of such an IUI. Others include [10], where songs are represented as discs that drop down from various taps (corresponding to different moods) and can be arranged and combined to form playlists. [21] and [22] present user interfaces to explore music collections according to different similarity dimensions (acoustic similarity as well as similarity derived from term profiles of artist-related Web pages). [25] proposes an interface that organizes a music collection in a large circular playlist by approximating the solution to a Traveling Salesman Problem that is defined by the collection's audio similarity matrix. Similar pieces of music are therefore found in similar regions along a disc visualization, which can be accessed via a wheel. Several extensions of this interface have been proposed, mainly to improve the user's orientation within the music collection: for example, [27] presents an implementation on a mobile device that incorporates Web-based tags to facilitate navigation, and [8] presents an approach that

[1]The majority of today's music playback devices still employs the traditional step-wise search scheme for `artist - album - track`.

ICMR '11, April 17-20, Trento, Italy

automatically structures the playlist hierarchically.

Most closely related to the deepTune interface is the nepTune application [13] that builds upon the "Islands of Music" (IoM) metaphor [19, 20]. According to this metaphor, similar music pieces are visualized via islands with homogeneous acoustic properties (for example, a "Classical" island, a "Heavy Metal" island, and so on). Depending on the (musical) distance between these islands, they are separated by large oceans or small sand banks. IoM uses the unsupervised learning algorithm *Self-Organizing Map* (SOM) [14] to determine music clusters and subsequently approximates the distribution of the collection's data items over the map by a *Smoothed Data Histogram* (SDH) [24]. In [13] a three-dimensional extension of the IoM is presented. The clusters are determined according to acoustic similarity. In addition, terms and images extracted from Web pages are presented to describe the regions of the map. While navigating through the landscape, the songs closest to the user's current position are played simultaneously. [17] presents a similar three-dimensional user interface. The authors, in contrast, use a metaphor different from the "Islands of Music". Their height map algorithm produces inverse heights, compared to the IoM approach, i.e., agglomerations of music pieces are located in valleys, and the clusters are not separated by oceans, but by hills. This technique resembles the *U-matrix* visualization [34] of the Self-Organizing Map. Moreover, user adaptation is supported by allowing the user to build or destroy separating hills. In this case, the similarity measure is adapted accordingly.

A drawback of the interface proposed in [13] is that it does not scale beyond some hundreds of songs, because of computational limitations and restricted visualization space. The deepTune application, in contrast, extends [13] in that it clusters the given music collection in a hierarchical manner, thus allows to visualize arbitrarily sized music collections. Given today's large amounts of tracks in personal music repositories, scalable, intelligent interfaces are of particular importance. We hence used a hierarchical clustering algorithm. The resulting multi-layer visualization requires various extensions to guide the user in her exploratory music discovery. These visual extensions, as well as the algorithm to find representative cluster prototypes that we had to elaborate, are detailed in the following section.

2. TECHNICAL FOUNDATIONS

The deepTune system makes use of various techniques from the fields of music information research, Web mining, and unsupervised learning. First, acoustic features (*Fluctuation Patterns*) are extracted from the audio signal of the input songs. Based on these features, a clustering algorithm then organizes the collection. Since deepTune should be able to visualize arbitrarily sized music collections, we opted for a hierarchical version of the Self-Organizing Map (SOM) clustering approach. Well-known prototypes for each cluster are subsequently determined by a Web-based popularity detection technique, the popularity ratings of which are combined with acoustic features in order to find music pieces that are representative for the cluster, but also popular enough to be of help for the majority of music listeners.

2.1 Signal-based Audio Features

Acoustic features are computed according to the approach proposed in [23]. The *Fluctuation Patterns* (FP) describe

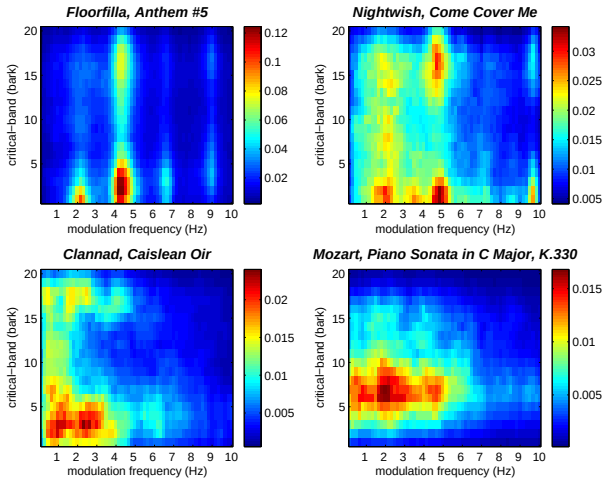


Figure 1: Fluctual Pattern visualization for different pieces of music.

rhythmical properties as they represent a music piece's distribution of re-occurring beats over different frequency bands and modulation frequencies (at different bpm).

To compute the FP features, first each track is sliced into 6-second-pieces. Then, the audio signal of selected pieces is transformed into the frequency domain by applying a *Fast Fourier Transform* (FFT) [6]. Subsequently, the frequency/magnitude scale is transformed into the *psychoacoustic Bark scale* [35], according to which frequency values are binned into perceptually equidistant "critical bands". Since the human ear is not equally responsive to all frequencies, a *perceptual model of the human auditory system* [31] is applied to account for the disproportionately high impact of low frequencies and the disproportionately low impact of high frequencies. Furthermore, *spectral masking* effects, i.e., the occlusion of quieter sounds if two or more sounds of similar frequency co-occur, are taken into account [28]. The modified Bark scale values are then transformed into the perceptually linear *Sone scale* [30]. The final *Fluctuation Pattern* of a piece of music is then obtained by computing the *Discrete Cosine Transform* (DCT) [1] of the modified power spectrum of each 6-second-slice, subsequent emphasizing perceptually important periodicities, and aggregating the resulting rhythm periodicity representations for the entire track by computing the median over all slices.

Figure 1 illustrates the Fluctuation Patterns of highly different music pieces. The FP depicted on the upper left belongs to a techno track with dominant bass beats around 120 and 240 bpm (corresponding to 2 and 4 Hertz, respectively). The piece on the lower left is a quiet song dominated by calm voices. The one on the upper right is an rock song with a poignant female voice. The piece on the lower right is a piano sonata with a clearly horizontal characteristic, without any activations in the lowest frequency bands.

Applying the FP computation results in a 1,200-dimensional feature vector representation (20 critical bands times 60 periodicity bins) for each piece of music. To decorrelate redundant feature dimensions and improve performance the feature vectors of all songs are compressed to 120 dimensions using *Principal Components Analysis* (PCA) [11]. This representation is then input into the clustering algorithm.

2.2 Clustering

To group similarly sounding music pieces (according to the Fluctuation Patterns), we reimplemented and extended the *Growing Hierarchical Self-Organizing Map* (GHSOM) clustering algorithm presented in [7]. The standard SOM is a neural network model that performs a non-linear mapping from a high-dimensional data space into a low-dimensional (usually two-dimensional) visualization space while preserving topological properties, i.e., data items that are similar in the feature space are mapped to similar positions of the visualization space. A SOM is described as a set of map units U arranged in a rectangle. Each map unit u_i is assigned a model vector m_i with same dimensionality as the data space. During training the model vectors are gradually adapted to better represent the input data X. The map unit's model vector closest to a data item x is referred to as x's "best-matching unit" and is used to represent x on the map.

In practice the standard SOM approach is limited in the number of data items that can be visualized. We therefore opted for the GHSOM approach that automatically adapts the structure of the SOM during training. Starting with a standard SOM of size 2×2, in each iteration step during training, the *mean quantization error* of each map unit mqe_i and of the whole SOM $mmqe$ is calculated according to Formulas 1 and 2, respectively. V_i represents the Voronoi set of map unit u_i, i.e., the set of all data items for which u_i is the best-matching unit, m_i is the model vector that describes u_i, and $|X|$ is the cardinality of the input data set.

$$mqe_i = \frac{1}{|V_i|} \cdot \sum_{j \in V_i} \|x_j - m_i\| \quad (1)$$

$$mmqe = \sum_i \frac{|V_i|}{|X|} \cdot mqe_i \quad (2)$$

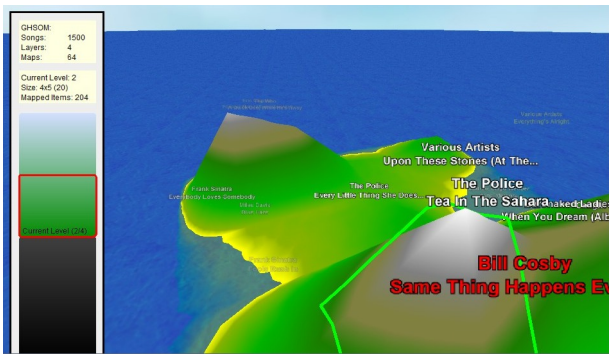
The parameter τ_m controls the size of individual SOMs, whereas τ_u regulates the depth of the GHSOM. In our experiments we empirically set $\tau_m = 0.5$ and $\tau_u = 0.25$.
To enforce a quadratic layout of the (sub-)SOMs, we further introduced a restricting parameter for the ratio between the number of rows and columns (set to 0.5). Moreover, we modified the algorithm in that SOMs representing less than 10 data items are not further expanded. This circumvents creating a lot of very sparse sub-level SOMs.

2.3 Cluster Prototype Selection

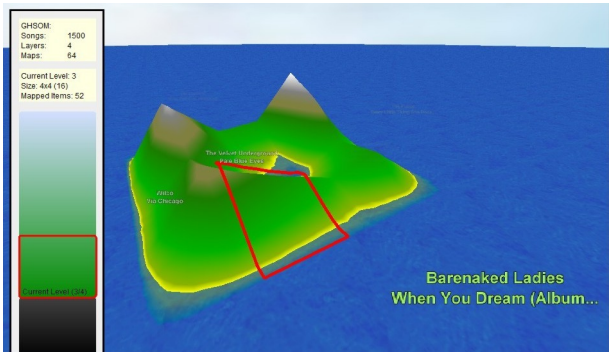
Depicting the labels of all music pieces mapped to the highest-level-SOM would yield tremendous visual overload when real-world collections consisting of tens of thousands of tracks are processed. An easy solution to this problem is to determine representative prototypes for each map unit u_i by selecting a number of data instances closest to m_i. Although this is a mathematically sound solution, the resulting prototypes are often not very popular, therefore unknown to most users, and thus of limited help for their orientation. As an alternative, we propose the following prototype selection algorithm that builds upon [8]. Prototypical music pieces for a map unit u_i are determined by combining Web-based popularity estimation and the pieces' audio-based distance to the respective map unit's model vector m_i.

First, we estimate the popularity of each artist in the collection by obtaining page counts from `Google` for queries

(a) top-level



(b) level two

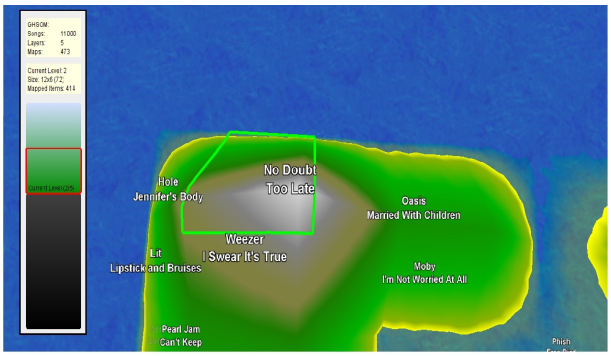


(c) level three

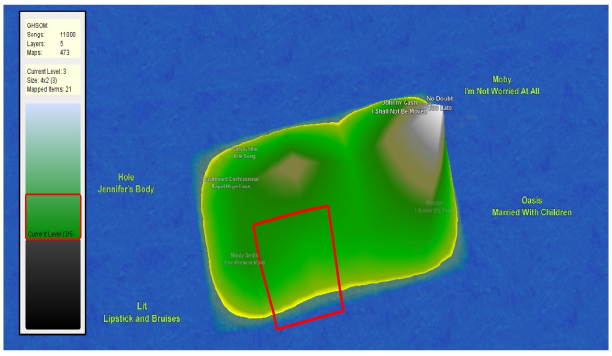
Figure 2: `deepTune`'s visualization on different levels of the GHSOM-tree.

of the form `"artist name" music review`. Based on the page-count-values, we define an artist ranking according to Formula 3, where $pc(a)$ is `Google`'s estimate for artist a's number of Web pages and $norm(\cdot)$ scales the values to the range $[1, 5]$. The audio signal-based part of the ranking function is given in Equation 4, where x is the feature vector corresponding to the music piece under consideration, $\|\cdot\|$ is the Euclidean distance, and $norm(\cdot)$ is a normalization function that shifts the range to $[1, 2]$. Finally, the artist-based popularity ranking and the track-based ranking of audio similarity to the model vector under consideration m_i are combined (Formula 5), and the pieces with highest $r(x)$ value are selected as prototypes for u_i.

$$r_w(a) = norm_{[1,5]}(log_{10}(pc(a))) \quad (3)$$



(a) upper level



(b) lower level

Figure 3: `deepTune` interface with anchor points.

$$r_s(x) = norm_{[1,2]}\left(\frac{1}{1+\ln(1+\|x-m_i\|)}\right) \quad (4)$$

$$r(x) = r_s(x) \cdot r_w(a) \quad (5)$$

An evaluation of this ranking technique has been conducted as well. The results are presented in Section 4.

3. USER INTERFACE

To get a first impression of the `deepTune` application, Figures 2(a), 2(b), and 2(c) depict the visualization resulting from a sample collection, respectively, on the top, on the second, and on the third level in the GHSOM-tree. The height of the landscape is derived from the voting matrix of the SDH [24], i.e., it roughly corresponds to the number of pieces mapped to each map unit. These height values are further encoded as colors, according to a color map used for topographical maps. The resulting landscape can be regarded as "Islands of Music". Each map unit is assigned a number of most representative tracks, the labels of which are depicted above the corresponding unit. Note that `deepTune` employs linear initialization of the SOM and batch training; the resulting maps are hence stable for a constant data set.

User interaction within the `deepTune` environment is provided either by using a mouse or a game pad. The interface supports panning, rotating, and adjusting the viewpoint angle. Furthermore, a "quick zoom" function facilitates swift orientation in large landscapes.

The currently played track is highlighted via flashing of its label. When moving through the landscape, a green rectangle around a map unit illustrates that the map unit can be expanded, i.e., lower-level SOMs do exist. A red rectangle denotes map units that cannot be expanded further. Upon pressing a button, the user "dives" into the surrounded map unit to the lower-level SOM. To prevent the user from getting lost in deep SOM hierarchies, sub-SOMs are placed into their higher-level context by showing the prototypes of their parents' neighboring map units, which serve as anchor points for better orientation. Figure 3 illustrates this concept by highlighting different anchor points (the green labels on the lower screenshot). Note that the layout of the anchor points within the lower level resembles the layout of the prototypes of the surrounding map units in the upper level (upper image). Moreover, a navigation bar illustrates the current depth in the GHSOM-tree and reveals further information on the visualization (e.g., the total number of SOMs and the size of the currently displayed SOM in terms of map units and represented data items). Furthermore, an "escape" function immediately brings the user back to the top-level SOM.

In order to alleviate the visual clutter that would arise from depicting the whole Voronoi set of each map unit, the number of prototypical pieces shown per map unit is limited. The actual number of prototypes shown for map unit u_i is determined by Equation 6, where V_i is the Voronoi set of map unit u_i, and m is the maximum number of prototypes per unit to be displayed.

$$np(u_i) = \left\lceil \frac{ln(|V_i|)}{ln(max_j(|V_j|))} \cdot m \right\rceil \quad (6)$$

3.1 Implementation Aspects

The audio features are calculated and compressed via PCA using the `CoMIRVA` framework [26] for music information retrieval and visualization. Also the artist popularity estimation builds upon Web retrieval functionality provided by `CoMIRVA`. We extended the framework by our variant of the GHSOM implementation.

The `deepTune` application itself is implemented in `Java`, using the libraries `Xith3D` for graphics processing and `OpenAL` as audio API. `deepTune` has been tested on a real-world music collection of about 48,000 songs, which is a subset of a digital music retailer's catalog.

4. EVALUATION

The selection of suitable cluster prototypes is essential for the usability of `deepTune`. To assess the quality of the proposed ranking approach (cf. Section 2.3), we compared the results obtained by our ranking function with play count data extracted from the music information system `last.fm` [15] for the same artists/tracks. To retrieve the play count data we used `last.fm`'s API [16]. Note that we refrained from directly using `last.fm`'s play counts in `deepTune` since building a Web crawler and simple page counts estimator is feasible without relying on commercial, proprietary systems.

Two evaluation steps have been performed. First, the pure Web-based artist ranking function $r_w(a)$ has been evaluated in order to assess its significance for the complete ranking function. Second, $r(x)$, the combined signal- and Web-based ranking function for particular sets of songs located on a specific map unit has been evaluated.

To illustrate the evaluation results, we calculated *Spearman's rank correlation coefficient* [29] and used a scatter plot.

4.1 Evaluation of $r_w(a)$

A list of 7,723 unique artist names has been extracted from our test database of 47,757 songs. For each artist name, two Web requests were issued. First, the `Google` page count corresponding to the query `"artist name" music review` was obtained. Second, the artist's overall `last.fm` play count was retrieved. Subsequently, tie-adjusted rankings were calculated. Tie adjustment was especially necessary for the calculation of Spearman's rank correlation coefficient, as uncorrected ties distort the result. Considering the page counts and play counts of the data set used, most ties were caused by either `Google` returning the value 0 for page counts or `last.fm` returning the value -1, indicating that the artist queried is known to the system. Tied ranks were dealt with by assigning each tied item the mean of its surrounding items' ranks.

Calculating Spearman's rank correlation coefficient for the tie-adjusted rankings results in the value 0.819, which indicates a strong correlation between `last.fm`'s play counts and `Google`'s page counts. This correlation is further revealed in the scatter plot depicted in Figure 4. Each point represents a specific artist, the axes correspond to the respective rankings. The x-axis represents the `Google`-based ranking, whereas the y-axis represents the ranking based on the `last.fm` play counts. The higher the value, the more popular an artist is considered by the respective data source. Thus, the highly unpopular artists should be located in the bottom left corner of the plot and the highly popular artists in the top right corner. If both rankings were perfectly similar, a straight line from the point of origin to the point (7,723; 7,723) would be visible. Even though this is obviously not the case, quite a strong correlation can be spotted, as most points are aligned around such an ideal line.

The upper left portion of the plot contains nearly no data points, contrary to the lower right portion. This indicates, that there are only few artists that have a low `Google` page count but a rather high `last.fm` playcount. Thus, if an artist is known to `Google`, he or she is very likely to be known to `last.fm` as well, but not necessarily vice versa. This finding can partly be traced back to misspellings as `Google` is more robust in that regard.

The straight horizontal and straight vertical sequences in the lower left portion of the plot are caused by the tie-corrected values. Both sequences indicate those artists that have either a page count value of zero or are unknown to `last.fm`. Such sequences affect Spearman's rank correlation coefficient positively. Nonetheless, when omitting both sequences in the calculation, a coefficient of 0.786 is attained, which is still very convincing.

In summary, a strong correlation between `Google` page counts, which serve as basis of `deepTune`'s prototypicality rating, and `last.fm` play counts could be determined.

4.2 Evaluation of $r(x)$

The setup for the evaluation of the track-based ranking function was quite similar to the evaluation of the artist ranking. When focusing on single tracks, however, the complete ranking function, i.e., the combination of signal- and Web-based ratings, has to be evaluated. Thus, instead of the cumulated artist play counts (over all tracks), we retrieved from `last.fm` the play counts of specific songs, i.e., combinations of `"artist name" - "track name"`.

First, the tie-corrected ranking function was applied to the

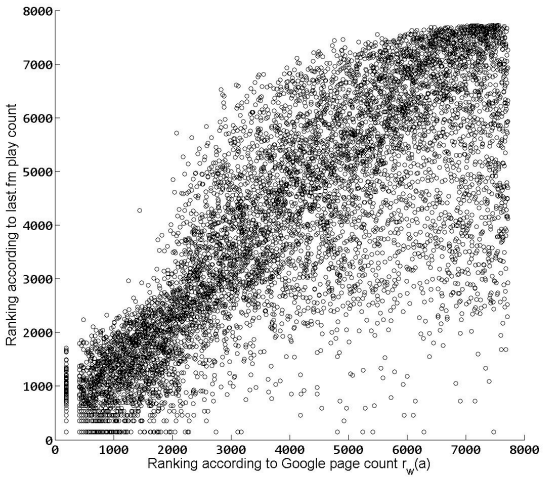


Figure 4: Scatter plot of artist ranking evaluation. Spearman's coefficient: 0.819**.**

ratings of each map unit's Voronoi set, and then the respective Spearman's rank correlation coefficient was calculated. Based on the entire collection of 47,757 songs, an overall measure as well as the results for two exemplary map units are discussed in the following.

To get a general impression of the quality of our prototype selection technique, the overall Spearman's rank correlation coefficient, averaged over all map units of the GHSOM's topmost map, was calculated according to Formula 7, where M is the number of map units, s_m is Spearman's rank correlation coefficient for map unit m having i_m data items mapped to, M denotes the total number of map units, and N the total number of data items.

$$s_{avg} = \frac{\sum_{m=1}^{M} s_m \cdot i_m}{N} \tag{7}$$

For the collection of 47,757 songs, our evaluation setting yielded an s_{avg} of 0.491, which states that the track-based ranking produced by `deepTune` correlates with the ranking of the corresponding `last.fm` play count values. Although this result is convincing, it is not as strong as the result of the pure artist-based evaluation. That is mostly due to the fact that the track rating also incorporates a signal-based component which does not necessarily correlate with the Web-based component.

Example 1 (Strong Correlation)

This example shows a comparison of the rankings of 1,312 songs drawn from a well-populated map unit. The correlation between both rankings becomes apparent from Figure 5. Spearman's rank correlation coefficient is 0.840, which indicates a strong positive correlation.

However, the long horizontal sequence of points in the lower left corner indicates a problem with `last.fm` not recognizing the queried artist/track, which may be traced back to misspellings. In addition, the collection contains some rather unknown songs. As such a sequence considerably affects Spearman's rank correlation coefficient, its calculation was repeated without those misspelled or unknown songs. Omitting those songs results in a smaller collection of 812

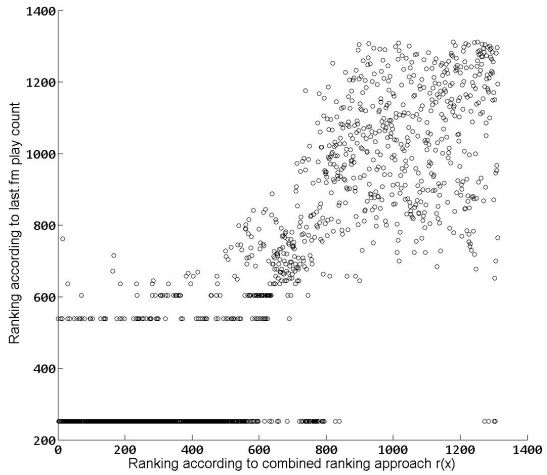


Figure 5: Track-based ranking evaluation of 1,312 **songs. Spearman's coefficient:** 0.840**.**

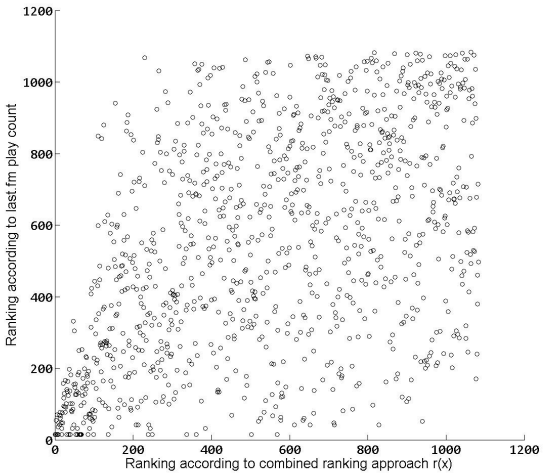


Figure 6: Track-based ranking evaluation of 1,083 **songs. Spearman's coefficient:** 0.497**.**

tracks. For this subset, Spearman's rank correlation coefficient is 0.751, which is still remarkable.

Example 2 (Fair Correlation)

This example shows the popularity rating of another map unit with a Voronoi set of cardinality 1,083. This time the results are not quite as clear as those of Example 1. Visually, the points of the respective scatter plot, depicted in Figure 6, appear much more widespread than in the previous example. Nonetheless, when examining the plot closely, some correlation can be identified, as more densely populated areas are located in the lower left and upper right corners whereas more scarcely populated areas are present in the upper left and lower right corners.

Spearman's rank correlation coefficient for this set is 0.497, a value that indicates some correlation, but is not as distinct and convincing as in the previous example.

Interestingly, the scatter plot reveals no prominent horizontal sequence of points at the bottom of the y-axis, which means that the Voronoi set of that particular map unit contains hardly any totally unknown or misspelled songs.

5. CONCLUSIONS AND FUTURE WORK

We presented a user interface to explore large music collections in virtual landscapes. Using a hierarchical clustering approach, we partition a music collection according to rhythmical features. We further proposed a method to determine meaningful cluster prototypes, and we implemented some techniques that facilitate the user's orientation within the hierarchical visualization framework.

Future work will include integrating automated playlist generation functionality. Moreover, we would like to extend the application by "social functions". For example, in a network version of `deepTune` each user could see the music currently listened to by her friends. Users may also be able to set visual markers or indicate favorite tracks or recommendations to other users. We are further assessing methods to port `deepTune` to mobile devices. Due to system and computing limitations of current mobile platforms, calculating the audio features will likely have to be carried out on a PC.

6. ACKNOWLEDGMENTS

This research is supported by the *Austrian Science Fund* (FWF): P22856-N23, L511-N15, and Z159.

7. REFERENCES

[1] N. Ahmed, T. Natarajan, and K. R. Rao. Discrete Cosine Transfom. *IEEE Transactions on Computers*, 23:90–93, January 1974.

[2] S. Baumann and O. Hummel. Using Cultural Metadata for Artist Recommendation. In *Proc. of WEDELMUSIC*, Leeds, UK, 2003.

[3] J. S. Breese, D. Heckerman, and C. Kadie. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In *Proc. of UAI*, San Francisco, USA, 1998. Morgan Kaufmann.

[4] O. Celma. *Music Recommendation and Discovery in the Long Tail*. PhD thesis, Universitat Pompeu Fabra, Barcelona, Spain, 2008.

[5] O. Celma, M. Ramírez, and P. Herrera. Foafing the Music: A Music Recommendation System Based on RSS Feeds and User Preferences. In *Proc. of ISMIR*, London, UK, 2005.

[6] J. W. Cooley and J. W. Tukey. An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, 19(90):297–301, 1965.

[7] M. Dittenbach, D. Merkl, and A. Rauber. The Growing Hierarchical Self-Organizing Map. In *Proc. of IJCNN*, Como, Italy, 2000.

[8] M. Dopler, M. Schedl, T. Pohle, and P. Knees. Accessing Music Collections via Representative Cluster Prototypes in a Hierarchical Organization Scheme. In *Proc. of ISMIR*, Philadelphia, USA, 2008.

[9] D. Eck, P. Lamere, T. Bertin-Mahieux, and S. Green. Automatic Generation of Social Tags for Music Recommendation. In *Proc. of NIPS*, 2008.

[10] M. Goto and T. Goto. Musicream: New Music Playback Interface for Streaming, Sticking, Sorting, and Recalling Musical Pieces. In *Proc. of ISMIR*, London, UK, 2005.

[11] I. T. Jolliffe. *Principal Component Analysis*. Springer, New York, USA, 1986.

[12] P. Knees, E. Pampalk, and G. Widmer. Artist Classification with Web-based Data. In *Proc. of ISMIR*, Barcelona, Spain, 2004.

[13] P. Knees, M. Schedl, T. Pohle, and G. Widmer. An Innovative Three-Dimensional User Interface for Exploring Music Collections Enriched with Meta-Information from the Web. In *Proc. of ACM Multimedia*, Santa Barbara, USA, 2006.

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

[15] http://last.fm (access: January 2010).

[16] http://last.fm/api (access: January 2010).

[17] D. Lübbers and M. Jarke. Adaptive Multimodal Exploration of Music Collections. In *Proc. of ISMIR*, Kobe, Japan, 2009.

[18] C. Mourlas and P. Germanakos, editors. *Intelligent User Interfaces*. Information Science Reference, Hershey, New York, USA, 2009.

[19] E. Pampalk. Islands of Music: Analysis, Organization, and Visualization of Music Archives. Master's thesis, Vienna University of Technology, Vienna, Austria, 2001.

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

[21] E. Pampalk and M. Goto. MusicRainbow: A New User Interface to Discover Artists Using Audio-based Similarity and Web-based Labeling. In *Proc. of ISMIR*, Victoria, Canada, 2006.

[22] E. Pampalk and M. Goto. MusicSun: A New Approach to Artist Recommendation. In *Proc. of ISMIR*, Vienna, Austria, 2007.

[23] E. Pampalk, A. Rauber, and D. Merkl. Content-based Organization and Visualization of Music Archives. In *Proc. of ACM Multimedia*, Juan les Pins, France, 2002.

[24] E. Pampalk, A. Rauber, and D. Merkl. Using Smoothed Data Histograms for Cluster Visualization in Self-Organizing Maps. In *Proc. of ICANN*, Madrid, Spain, 2002.

[25] T. Pohle, P. Knees, M. Schedl, E. Pampalk, and G. Widmer. "Reinventing the Wheel": A Novel Approach to Music Player Interfaces. *IEEE Transactions on Multimedia*, 9, 2007.

[26] M. Schedl, P. Knees, K. Seyerlehner, and T. Pohle. The CoMIRVA Toolkit for Visualizing Music-Related Data. In *Proc. of EuroVis*, Norrköping, Sweden, 2007.

[27] D. Schnitzer, T. Pohle, P. Knees, and G. Widmer. One-Touch Access to Music on Mobile Devices. In *Proc. of MUM*, Oulu, Finland, 2007.

[28] M. R. Schröder, B. S. Atal, and J. L. Hall. Optimizing Digital Speech Coders by Exploiting Masking Properties of the Human Ear. *Journal of the Acoustical Society of America*, 66:1647–1652, 1979.

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

[30] S. S. Stevens. A Scale for the Measurement of the Psychological Magnitude: Loudness. *Psychological Review*, 43(5):405–416, 1936.

[31] E. Terhardt. Calculating Virtual Pitch. *Hearing Research*, 1:155–182, 1979.

[32] D. Turnbull, R. Liu, L. Barrington, and G. Lanckriet. A Game-based Approach for Collecting Semantic Annotations of Music. In *Proc. of ISMIR*, Vienna, Austria, 2007.

[33] R. C. Veltkamp. Multimedia Retrieval Algorithmics. In *Proc. of the SOFSEM)*, Harrachov, Czech Republic, 2007.

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

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

Markus Schedl, Dominik Schnitzer

Hybrid Retrieval Approaches to Geospatial Music Recommendation

Proceedings of the 35th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)

Dublin, Ireland, July–August 2013

Hybrid Retrieval Approaches to Geospatial Music Recommendation

Markus Schedl
Department of Computational Perception
Johannes Kepler University, Linz, Austria
markus.schedl@jku.at

Dominik Schnitzer
Austrian Research Institute for Artificial Intelligence, Vienna, Austria
dominik.schnitzer@ofai.at

ABSTRACT

Recent advances in music retrieval and recommendation algorithms highlight the necessity to follow multimodal approaches in order to transcend limits imposed by methods that solely use audio, web, or collaborative filtering data. In this paper, we propose hybrid music recommendation algorithms that combine information on the *music content*, the *music context*, and the *user context*, in particular, integrating location-aware weighting of similarities. Using state-of-the-art techniques to extract audio features and contextual web features, and a novel standardized data set of music listening activities inferred from microblogs (`MusicMicro`), we propose several multimodal retrieval functions.

The main contributions of this paper are (i) a systematic evaluation of mixture coefficients between state-of-the-art audio features and web features, using the first standardized microblog data set of music listening events for retrieval purposes and (ii) novel geospatial music recommendation approaches using location information of microblog users, and a comprehensive evaluation thereof.

1. INTRODUCTION

The field of Music Information Retrieval (MIR) is seeing a paradigm shift, away from system-centric perspectives towards user-centric approaches [3]. In this vein, incorporating user models and addressing user-specific demands in music retrieval and music recommendation systems is becoming more and more important.

We present several approaches that combine *music content*, *music context*, and *user context* aspects to build a hybrid music retrieval system [12]. Music content and music context are incorporated using state-of-the-art feature extractors and corresponding similarity estimators. The user context is addressed by taking into account *musical preference* and *geospatial data*, using a standardized collection of listening behavior mined from microblog data [11].
We make use of the best feature extraction and similarity computation algorithms currently available to model *music content* and *music context*. We then integrate these similarity models as well as a *user context* model into a novel user-aware music recommendation approach that encompasses all three modalities important to human music perception [12].

The main contributions of this paper are: (i) a systematic evaluation of combining audio- and web-based state-of-the-art approaches to music similarity measurement and (ii) two approaches to incorporate geospatial information into music recommendation algorithms.

The remainder of the paper is organized as follows. Section 2 details the acquisition of the raw music (meta-)data, which serves as input to the feature extraction and data representation techniques presented in Section 3. In Section 4, we construct different hybrid (music content and music context) models and systematically evaluate their mixture coefficients. Section 5 then proposes two methods to incorporate geospatial information into music recommendation models. These extended models are evaluated and compared to the respective models without geospatial data and to a random baseline. Section 6 briefly reviews related literature. Eventually, Section 7 draws conclusions and points to further research directions.

2. DATA ACQUISITION

The only standardized public data set of microblogs, as far as we are aware of, is the one used in the TREC 2011 and 2012 Microblog tracks[1] [4]. Although this set contains approximately 16 million tweets, it is not suited for our task as it is not tailored to music-related activities, i.e. the amount of music-related posts is marginal.

We hence have to acquire multimodal data sets of *music items* and *listeners*, reflecting the three broad aspects of human music perception (*music content*, *music context*, and *user context*) [12]. Whereas the *music content* refers to all information that is derived from the audio signal itself (such as ryhthm, timbre, or melody), the *music context* covers contextual information that cannot be derived from the actual audio with current technology (e.g., meaning of song lyrics, background of a performer, or co-listening relationships between artists). The *user context* encompasses all information that are intrinsic to the listener. Examples range from musical education to spatiotemporal properties to physiological measures to current activities.

User Context.

Only very recently a data set of music listening activities inferred from microblogs has been released [11]. It is en-

[1] `http://trec.nist.gov/data/tweets`

SIGIR 2013 Dublin, Ireland

titled MusicMicro and is freely available[2], fostering reproducibility of social media-related MIR research. This data set contains about 600,000 listening events posted on Twitter[3]. Each event is represented by a tuple <*twitter-id, user-id, month, weekday, longitude, latitude, country-id, city-id, artist-id, track-id*>, which allows for spatiotemporal identification of listening behavior.

Music Content.

Based on the lists of artist and song names in the MusicMicro collection, we gather snippets of the songs from 7digital[4]. These serve as input to the music content feature extractors (cf. Section 3).

Music Context.

To capture aspects of human music perception which are not encoded in the audio signal, we extract music-related web pages that represent such contextual information. Following the approach suggested in [13], we retrieve the top 50 web pages returned by the Bing[5] search engine for queries comprising the artist name[6] and the additional keyword "music", to disambiguate the query for artists such as "Bush", "Kiss", or "Hole".

In summary, we gathered raw data covering each of the three categories of perceptual music aspects [12]: *music content* (audio snippets), *music context* (related web pages), and *user context* (user-specific music listening events with spatiotemporal labels).

3. DATA REPRESENTATION

To represent the *music content*, we use state-of-the-art audio music feature extractors proposed in [7]. These algorithms won three times in a row (since 2010) the annually run benchmarking activity *Music Information Retrieval Evaluation eXchange* (MIREX): "Audio Music Similarity and Retrieval" task[7]. They hence constitute the reference in music feature extraction for similarity-based retrieval tasks. More precisely, we extract the auditory music features proposed in [7], which combine various rhythmic features derived from the audio signal, e.g., "onset patterns" and "onset coefficients" (note onsets), with timbral features, e.g., "Mel Frequency Cepstral Coefficients" (coarse description of the amplitude envelop). The eventual output is pairwise similarity estimates between songs, which are later aggregated to the artist level.

We again employ a state-of-the-art technique to obtain features reflecting the *music context*. To describe the music items at the artist level, we follow the approach proposed in [13]. In particular, we model each artist by creating a "virtual artist documents", i.e. we concatenate all web pages retrieved for the artist. In accordance with findings of [10], we then use a dictionary of music-related terms (genres, styles, instruments, and moods) to index the resulting documents. From the index, we compute term weights according to the best feature combination found in the large-scale experiments of [13]: TF_C3.IDF_I.SIM_COS, i.e. computing term weight vectors and artist similarity estimates according to Equations 1, 2, and 3, respectively for *tf*, *idf*, and *cosine similarity*; $f_{d,t}$ represents the number of occurrences of term t in document d, N is the total number of documents, $\mathcal{D}_t$ is the set of documents containing term t, F_t is the total number of occurrences of term t in the document collection, $\mathcal{T}_d$ is the set of distinct terms in document d, and W_d is the length of document d.

$$tf_{d,t} = 1 + \log_2 f_{d,t} \tag{1}$$

$$w_t = 1 - \frac{n_t}{\log_2 N}, \quad n_t = \sum_{d \in \mathcal{D}_t} \left(-\frac{f_{d,t}}{F_t} \log_2 \frac{f_{d,t}}{F_t} \right) \tag{2}$$

$$S_{d_1,d_2} = \frac{\sum_{t \in \mathcal{T}_{d_1,d_2}} (w_{d_1,t} \cdot w_{d_2,t})}{W_{d_1} \cdot W_{d_2}} \tag{3}$$

3.1 Availability of the Data Sets

All components of the data set used in this paper are publicly available to allow researchers reproduce the results reported. The sole exception is the actual audio content of the songs under consideration. We cannot share them due to copyright restrictions. However, we provide identifiers by means of which corresponding 30-second-clips can be downloaded from 7digital. If you are interested in the data sets, please contact the first author.

4. HYBRID MUSIC RETRIEVAL

One main research question is how to ideally combine audio and web features for music retrieval. Although quite a few MIR researchers suggest such a combination [2, 1, 3, 12, 5], a systematic evaluation of combining state-of-the-art audio and web similarity estimators is still missing, hence provided here.

4.1 Experimental Setup

In a preprocessing step, we aggregate the audio features on the artist level, as they are computed on single tracks. To obtain audio similarities $asim(i,j)$ between two artists i and j, we compute the minimum of the distances between all pairs of tracks by i and j as the minimum yielded the best results in preliminary experiments similar to the ones described later in this section. Web similarities $wsim(i,j)$ are already defined on the artist level. Both, audio and web similarites, are normalized using the global distance scaling method Mutual Proximity [14].

Linear combinations of web similarities and audio similarities yield a hybrid similarity function $sim(i,j)$ between artists i and j. It is given in Equation 4, where ξ is the mixture coefficient, i.e., the weight of the audio part, different values of which we systematically evaluate.

$$sim(i,j) \; = \; \xi \cdot asim(i,j) \; + \; (1-\xi) \cdot wsim(i,j) \tag{4}$$

As *gold standard* we use genre information and assess retrieval performance via the overlap between the genres assigned to the query artist and those assigned to his K nearest neighbors according to the similarity function under investigation. This is a standard evaluation approach in MIR. We gather genre information by (i) retrieving the top tags for each artist via the Last.fm API[8] and (ii) using the top

[2] http://www.cp.jku.at/musicmicro
[3] http://www.twitter.com
[4] http://www.7digital.com
[5] http://www.bing.com
[6] Please note that issuing queries at the song level is not reasonable, as doing so typically yields only very few results.
[7] http://www.music-ir.org/mirex/wiki/2012:Audio_Music_Similarity_and_Retrieval
[8] http://www.last.fm/api

ξ	$K=1$	$K=3$	$K=5$
web only – 0.00	**.5829**	**.5753**	**.5774**
.05	.6421	.6280	.6257
.15	.6432	**.6286**	**.6261**
.25	**.6433**	.6275	.6258
.35	.6430	.6275	.6257
.45	.6408	.6266	.6252
.55	.6394	.6259	.6244
.65	.6379	.6255	.6232
.75	.6368	.6234	.6221
.85	.6330	.6202	.6188
.95	.6215	.6083	.6059
audio only – 1.00	**.5436**	**.5302**	**.5247**

Table 1: Overlap scores for different mixture coefficients ξ between web and audio features.

20 main genres from `allmusic`[9] to index the sets of tags retrieved.

To evaluate retrieval performance, we use a Jaccard-like overlap measure, shown in Equations 5 and 6, where i is the query artist, $Genres_i$ is the set of genres assigned to i, K is the number of i's nearest neighbors to consider, and A is the number of all artists in the data set. The range of the performance measures is $[0, 1]$, i.e., they are 1.0 if the genres of the seed artist i's K nearest neighbors perfectly overlap with those of i.

$$overlap_i = \frac{1}{K} \cdot \sum_{j=1\ldots K} \frac{|Genres_i \cap Genres_j|}{|Genres_i|} \quad (5)$$

$$overlap = \frac{1}{A} \cdot \sum_{i=1\ldots A} overlap_i \quad (6)$$

4.2 Results

Performance scores for the hybrid retrieval function for different mixture coefficients ξ are shown in Table 1, together with results for a random baseline. Although using only web features ($\xi = 0.0$) yields better results than using audio only ($\xi = 1.0$), adding a small amount of content features to web features (or vice versa) boosts performance considerably. Adding a small amount of a complementary similarity component thus proves highly beneficial. Overall, values of ξ around 0.15 perform best. We hence use Equation 7 as hybrid (audio and web features) music model (MU) for subsequent experiments.

$$sim(i,j) = 0.15 \cdot asim(i,j) + 0.85 \cdot wsim(i,j) \quad (7)$$

5. MUSIC RECOMMENDATION MODELS

Building recommendation systems requires a user model. In our case, each user u is modeled by the set of artists $UM(u)$ he listened to. Based on this simple model, we implement the following recommendation strategies: (i) the hybrid music retrieval model (MU) elaborated in the previous section and (ii) a standard collaborative filtering (CF) model. In the MU model, the hybrid music similarity function (Equation 7) is used to determine the artists closest to $UM(u)$, which are then recommended. In the CF model, the users closest to u are determined (using the Jaccard index between the user models), and the artists listened to by these nearest users are recommended. For comparison, we further implemented a random baseline model (BL) that randomly picks K users from the filtered user set (via the parameter τ, see below) and recommends the artists they listened to. To integrate *geospatial information* into the CF model, we first compute a centroid of each user u's geospatial listening distribution $\mu_u[\lambda, \varphi]$[10]. We then use the normalized geodesic distance $gdist(u, v)$ (Equation 8) between the seed user u and each other user v to weight the distance based on the user models. To this end, we propose two different weighting schemes: linear weighting and weighting according to a Gaussian kernel around $\mu_u[\lambda, \varphi]$. We eventually obtain a geospatially modified user similarity $sim(u, v)$ by adapting the Jaccard index between $UM(u)$ and $UM(v)$ via geospatial, linear or Gauss weighting, according to Equation 9 (GEO-Lin) or Equation 10 (GEO-Gauss), respectively. We recommend the artists listened to by u's nearest users v. Table 2 summarizes all recommendation algorithms under investigation.

Abbreviation	Description
BL	random baseline
MU	hybrid music model (Equation 7)
CF	collaborative filtering model
CF-GEO-Lin	CF model: geospatial user weighting using linear spatial distances
CF-GEO-Gauss	CF model: geospatial user weighting weighting using a Gauss kernel

Table 2: Overview of recommendation models.

$$gdist(u,v) = \arccos\big(\sin(\mu_u[\varphi]) \cdot \sin(\mu_v[\varphi]) + \cos(\mu_u[\varphi]) \cdot \cos(\mu_v[\varphi]) \cdot \cos(\mu_u[\lambda] - \mu_v[\lambda])\big) \cdot \max(gdist)^{-1} \quad (8)$$

$$sim(u,v) = J(UM(u), UM(v)) \cdot gdist(u,v)^{-1} \quad (9)$$

$$sim(u,v) = J(UM(u), UM(v)) \cdot \exp(-gdist(u,v)) \quad (10)$$

5.1 Experimental Setup

In order to ensure sufficient artist coverage of users, we evaluate our models using different thresholds τ for the minimum number of unique artists a user must have listened to in order to include him in the experiments. We vary τ between 50 and 150 using a step size of 10. Denoting as U_τ the number of users in the `MusicMicro` data set with equal or more than τ unique artists, we perform U_τ-fold leave-one-out cross-validation for each value of τ.

5.2 Results

Figure 1 shows accuracies for $K = [3, 5]$ nearest neighbors and $\tau = [50 \ldots 150]$. We can see that all approaches significantly outperform the random baseline. Comparing the MU approach with the CF approaches, it is evident that CF generally works better for data sets with high numbers of users (smaller τ), while content-based MU outperforms CF when the number of users is restricted. This finding suggests a combination of MU and CF, which will be addressed as part of future work. As for geospatial weighting, a similar

[9] `http://www.allmusic.com`

[10] It is common to denote longitude by λ and latitude by φ.

observation comparing the linear weighting with the Gauss weighting can be made. The more active the users (higher τ), the better the performance of the linear weighting approach, and the worse the Gauss kernel approach. An explanation for this may be that very frequent users of `Twitter` typically live in agglomerations, whereas occasional twitterers live in less densely populated areas. For these users in rural areas, a Gauss weighting is seemingly beneficial as very nearby users frequently know each other and share common music tastes (which is not true for highly populated areas). The models that integrate geospatial information outperform the standard CF model for high τ values, indicating again that this kind of information is particular beneficial for "power users", who typically live in densely populated areas.

6. RELATED WORK

Specific related work on geospatial music retrieval is very sparse, probably due to the fact that geospatially annotated music listening data is hardly available. Among the few works, Park et al. [6] use geospatial positions and suggest music that matches a selected environment, based on aspects such as ambient noise, surrounding, or traffic. Raimond et al. [8] combine information from different sources to derive geospatial information on artists, aiming at locating them on a map. Zangerle et al. [15] use a co-occurrence-based approach to map tweets to artists and songs and eventually construct a music recommendation system. However, they do not take location into account.

On a more general level, this work relates to context-based and hybrid recommendation systems, a detailed review of which is unfortunately beyond the scope of the paper. A comprehensive elaboration, including a decent literature overview, can be found in [9].

7. CONCLUSIONS AND OUTLOOK

We presented the first *systematic evaluation of hybrid music retrieval approaches* (combining the currently best performing audio/music content and web/music context features), using a recently published, standardized data set of music listening activities mined from microblogs. Experiments showed that a linear mixture coefficient of 0.15 for the audio part and 0.85 for the web component performed best, overall. Interestingly, adding only a very small amount of audio-based information to web features (or vice versa) considerably improves results.

To the best of our knowledge, this is also the first work that *integrates geospatial information into music recommendation algorithms*. Experiments indicate that including geospatial information is particularly beneficial for music recommendation when users listen to many different artists. The collaborative filtering approach (CF) outperforms the hybrid music retrieval model (MU) when the data set comprises a high number of users who listen to less artists, overall.

Future work will include considering more diverse data about the user context, such as demographics, listening time (hour of day, working day versus weekend), or gender. In addition, we plan to combine the MU and the CF models, including geospatial weighting. As a further usage scenario, we target users frequently traveling around the world and wanting to listen to music tailored to their current location, but also complying to their music taste. We will look into adapting our approaches accordingly.

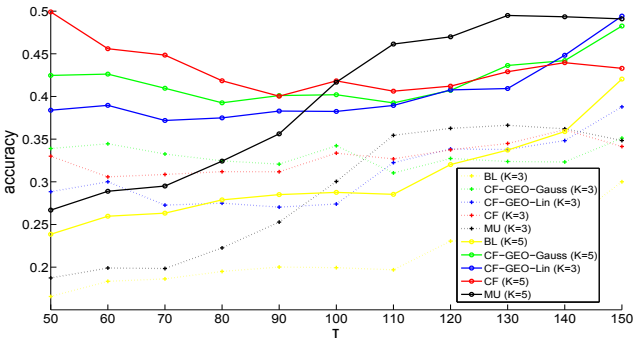


Figure 1: Accuracy plots for different values of K and τ.

Acknowledgments

This research is supported by the Austrian Science Funds (FWF): P22856, P24095, P25655, and by the EU FP7/2007-2013 through the PHENICX project under grant agreement no. 601166.

8. REFERENCES

[1] D. Bogdanov, J. Serrà, N. Wack, P. Herrera, and X. Serra. Unifying Low-Level and High-Level Music Similarity Measures. *IEEE Transactions on Multimedia*, 13(4):687–701, Aug 2011.

[2] E. Coviello, A. B. Chan, and G. Lanckriet. Time Series Models for Semantic Music Annotation. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(5):1343–1359, Jul 2011.

[3] C. Liem, M. Müller, D. Eck, G. Tzanetakis, and A. Hanjalic. The Need for Music Information Retrieval with User-centered and Multimodal Strategies. In *Proc. MIRUM*, Scottsdale, AZ, USA, 2011.

[4] R. McCreadie, I. Soboroff, J. Lin, C. Macdonald, I. Ounis, and D. McCullough. On Building a Reusable Twitter Corpus. In *Proc. SIGIR*, Portland, OR, USA, 2012.

[5] B. McFee and G. Lanckriet. Heterogeneous Embedding for Subjective Artist Similarity. In *Proc. ISMIR*, Kobe, Japan, 2009.

[6] S. Park, S. Kim, S. Lee, and W. S. Yeo. Online Map Interface for Creative and Interactive MusicMaking. In *Proc. NIME*, Sydney, Australia, 2010.

[7] T. Pohle, D. Schnitzer, M. Schedl, P. Knees, and G. Widmer. On Rhythm and General Music Similarity. In *Proc. ISMIR*, Kobe, Japan, 2009.

[8] Y. Raimond, C. Sutton, and M. Sandler. Automatic Interlinking of Music Datasets on the Semantic Web. In *Proc. WWW: LDOW Workshop*, Beijing, China, 2008.

[9] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, editors. *Recommender Systems Handbook*. Springer, 2011.

[10] M. Schedl. #nowplaying Madonna: A Large-Scale Evaluation on Estimating Similarities Between Music Artists and Between Movies from Microblogs. *Information Retrieval*, 15:183–217, June 2012.

[11] M. Schedl. Leveraging Microblogs for Spatiotemporal Music Information Retrieval. In *Proc. ECIR*, Moscow, Russia, 2013.

[12] M. Schedl and A. Flexer. Putting the User in the Center of Music Information Retrieval. In *Proc. ISMIR*, Porto, Portugal, 2012.

[13] M. Schedl, T. Pohle, P. Knees, and G. Widmer. Exploring the Music Similarity Space on the Web. *ACM Transactions on Information Systems*, 29(3), Jul 2011.

[14] D. Schnitzer, A. Flexer, M. Schedl, and G. Widmer. Local and global scaling reduce hubs in space. *Journal of Machine Learning Research*, 13:2871–2902, October 2012.

[15] E. Zangerle, W. Gassler, and G. Specht. Exploiting Twitter's Collective Knowledge for Music Recommendations. In *Proc. WWW: #MSM Workshop*, Lyon, France, 2012.