Chapter 13
Music Recommender Systems

Markus Schedl, Peter Knees, Brian McFee, Dmitry Bogdanov, and Marius Kaminskas

13.1 Introduction

Boosted by the emergence of online music shops and music streaming services, digital music distribution has led to an ubiquitous availability of music. Music listeners, suddenly faced with an unprecedented scale of readily available content, can easily become overwhelmed. Music recommender systems, the topic of this chapter, provide guidance to users navigating large collections. Music items that can be recommended include artists, albums, songs, genres, and radio stations.

In this chapter, we illustrate the unique characteristics of the music recommendation problem, as compared to other content domains, such as books or movies. To understand the differences, let us first consider the amount of time required for a user to consume a single media item. There is obviously a large discrepancy in consumption time between books (days or weeks), movies (one to a few hours), and a song (typically a few minutes). Consequently, the time it takes for a user to form opinions for music can be much shorter than in other domains, which contributes to the ephemeral, even disposable, nature of music. Similarly, in music, a single

M. Schedl (✉) • P. Knees
Department of Computational Perception, Johannes Kepler University Linz, Linz, Austria
e-mail: markus.schedl@jku.at; peter.knees@jku.at

B. McFee
Center for Data Science, New York University, New York, NY, USA
e-mail: brian.mcfee@nyu.edu

D. Bogdanov
Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain
e-mail: dmitry.bogdanov@upf.edu

M. Kaminskas
Insight Centre for Data Analytics, University College Cork, Cork, Ireland
e-mail: marius.kaminskas@insight-centre.org

F. Ricci et al. (eds.), *Recommender Systems Handbook*,
DOI 10.1007/978-1-4899-7637-6_13

item may be consumed repeatedly (even multiple times in a row), while other media items are typically consumed at most a few times. This implies that a user might not only tolerate, but actually appreciate recommendations of already known items.

On a practical level, another distinguishing property is that music can be directly addressed at different levels of abstraction. For instance, while movie recommenders typically suggest individual items to the user, music recommendation approaches may suggest groupings of items by genre, artist, or albums.

From a practitioner's perspective, we note that collaborative filtering techniques are inherently domain-agnostic, and can be easily applied to music rating data [131, 134].[1] However, in the music domain, explicit rating data is relatively rare, and even when available, tends to be sparser than in other domains [44]. Instead, implicit positive feedback is often drawn from uninterrupted (or unrejected) listening events.

Due to the sparsity of readily available user feedback data, music recommendation techniques tend to rely more upon content descriptions of items than techniques in other domains. Content-based music recommendation techniques are strongly tied to the broader field of music information retrieval (MIR), which aims at extracting semantic information from or about music at different representation levels (e.g., the audio signal, artist or song name, album cover, or score sheet).[2] Many of these approaches apply signal processing and analysis methods directly to music in order to extract musically meaningful features and in turn enable novel search and browsing interfaces. In all these scenarios, as is the case with memory-based collaborative filtering methods (see Chap. 2), the concept of similarity is central. For content-based approaches, item similarity is typically computed between item feature vectors. Section 13.2 provides an overview of content-based music recommendation techniques, including both metadata and signal analysis methods.

From the user's perspective, content can play an important role in influencing preferences for music. Studies in music psychology show that a user's short-term music preferences are influenced by various factors, such as the environment, the emotional state, or the activity of the user [97]. We elaborate on contextual music recommendation approaches in Sect. 13.3. In Sect. 13.4, we present hybrid recommendation approaches which combine collaborative filtering, content-based, and context-based methods.

Because users often listen to several songs in rapid succession—e.g., via streaming radio or a personal music device—some recommender systems have been designed specifically for serial recommendation [59]. Due to the unique

[1] We will not further detail collaborative filtering of music ratings in this chapter. To understand the principles of this technique, we refer the reader to Chap. 2.

[2] To avoid confusion, we note that *content* has different connotations within the MIR and recommender systems communities. MIR makes an explicit distinction between (content-based) approaches that operate directly on audio signals and (metadata) approaches that derive item descriptors from external sources, e.g., web documents [70]. In recommender systems research, as in the remainder of this chapter, both types of approaches are described as "content-based".

constraints and modeling assumptions of serial consumption, the evaluation criteria and algorithmic solutions diverge substantially from the more standard techniques found in the recommender systems literature. Section 13.5 provides an overview of automatic playlist generation, including algorithms and evaluation methodologies.

In Sect. 13.6, we discuss common evaluation strategies, benchmarking campaigns, and data sets used in music recommendation research. Finally, we conclude by highlighting current research challenges in Sect. 13.7.

13.2 Content-Based Music Recommendation

Content information includes any information describing music items that can be extracted from the audio signal, as well as metadata provided by external sources (e.g., web documents, discography data, or tags). In this section, we overview research on content-based approaches to music recommendation, and categorize the existing approaches with respect to the employed information sources.

13.2.1 Metadata Content

Musical metadata comes in several forms, including *manual annotations* provided by experts, *social tags* obtained from collaborative tagging services, and annotations *automatically mined from the web* using text retrieval techniques. Although some studies have demonstrated such metadata may not perform as well as collaborative filtering techniques [54], it can be used to augment or replace collaborative filtering in cold-start scenarios [19, 84].

13.2.1.1 Manual Annotations

Manual annotations include editorial metadata, such as musical genre and subgenre, record label, year and country of release, relations between artists, tracks, and albums, as well as any other associated production information. Additionally, annotations of musical properties such as tempo, mood, and instrumentation can be used to provide detailed summaries of musical content.

There is a number of online databases for editorial metadata, which are built by either music experts or moderated communities of enthusiasts. These databases ensure a certain quality of data, but impose limitations on its structure, e.g., by adhering to genre taxonomies [101]. *MusicBrainz*[3] and *Discogs*[4] provide extensive,

[3] http://www.musicbrainz.org.

[4] http://www.discogs.com.

freely available, community-built information on artists, record labels, and their releases. This information is related to the cultural context of music, but it omits annotations of detailed musical properties beyond genre and musical epoch (e.g., 90s). Although limited, editorial metadata has been used to build simple genre-based recommenders [82], to refine audio content-based methods (e.g., [18]; cf. Sect. 13.2.2), or in hybrid recommenders (e.g., [25]; cf. Sect. 13.4).

Bogdanov et al. [19] build an artist recommender exclusively using metadata from the *Discogs* database. For each artist in the database, a tag weight vector is created by propagating genre, style, record label, country, and year information for each release related to the artist. Relations between artists (aliases, membership in groups) and the role of the artist in each associated release—e.g., main artist, remixing/performing credits on a release, etc.—are taken into account. Artist similarity is measured by comparing sparse tag weight vectors, which are compressed using latent semantic analysis (LSA) [37].

Manual annotations of properties other than genre and epoch are promising, but they are more costly, and difficult to scale to large collections. *Pandora*[5] is an example of a large-scale commercial recommender system using such annotations done by experts [67]. Similarly, *AllMusic*[6] is an example of a commercial database that provides mood annotations in addition to general editorial metadata. However, relatively few academic studies incorporate these manual annotations because they are proprietary, and no public data sets of this kind (and scale) are available for researchers. Existing work therefore resorts to individual, hand-made annotations, for instance of genre, tempo, mood [105, 139], year [139], and emotion [81].

13.2.1.2 Social Tags

In contrast to structured taxonomy-driven expert annotations, information about music items can be collected from social tagging services. Social tagging services allow casual users to provide unstructured text annotations for any item. Social tags, while inherently noisy, can draw from a larger pool of annotators, and noisy annotations can be combined to derive a structured *folksonomy* of tags [135]. The *Last.fm*[7] music tagging service has gained some popularity in academic research by providing open access to an extensive collection of music tags. It includes uncategorized tags describing genres, moods, instrumentation, and locations, as well as personal associations evoked in the users by music (e.g., *favorite* or *seen live*) [58]. The tags can be easily obtained for particular artists or tracks, which can be used to assess similarity between items by comparing respective tag weight vectors [54]. Similarity comparisons can be enhanced by latent semantic analysis techniques to overcome the problem of vector sparsity [74].

[5]http://www.pandora.com.

[6]http://www.allmusic.com.

[7]http://www.last.fm.

13.2.1.3 Annotations by Web Content Mining

As an alternative to social tags, keyword annotations can be mined from music-related web pages using text processing techniques. Keywords can be extracted from web pages, blogs and RSS feeds related to music items, as well as lyrics databases. Schedl et al. provide an overview of text mining techniques for measuring artist similarity [123], and create a large-scale music search system which operates on an index of artist term profiles [126]. A similar approach by Barrington et al. [140] limits the keyword mining process to specific web sites with high-quality music information, such as *AllMusic*, *Wikipedia*,[8] *Amazon*,[9] *BBC*,[10] *Billboard*,[11] or *Pitchfork*.[12] An early study by Pazzani and Billsus [108] describes a recommendation approach which used a naïve Bayes classifier to predict user preferences from artist keywords extracted from web pages. Green et al. [54] retrieve keywords from *Wikipedia* artist entries and social tags from *Last.fm*. They propose to generate recommendations based on artist-to-artist similarity, or similarity between artists and a vector of keyword weights summarizing the user's favorite artists. Similarly, McFee and Lanckriet [88] combine social tags and keywords extracted from artist biographies found on *Last.fm* to predict artist similarity ratings. Celma et al. [35] extract keywords from RSS feeds related to music artists, and then generate recommendations by ranking artists by similarity to a set of preferred artists. Finally, Lim et al. [77] learn a song-level similarity function from topic models over bag-of-words representations of lyrics provided by *musiXmatch.com*.

13.2.2 Audio Content

Audio content analysis is advocated by MIR researchers as an alternative or complement to metadata and collaborative filtering methods [12, 29]. Recommender systems based on audio content are not susceptible to popularity bias, and are therefore expected to reveal the "long tail" of music consumption [31]. Music descriptors obtained by audio signal analysis can enhance music search by enabling novel ways for querying and interacting with music collections.

Audio content analysis can provide various types of information which can be incorporated in recommender systems. This information can be broadly divided into two categories: acoustic and musical features computed directly from audio, and semantic annotations inferred or predicted from these acoustic features by machine learning techniques.

---

[8] http://www.wikipedia.org.

[9] http://www.amazon.com.

[10] http://www.bbc.co.uk.

[11] http://www.billboard.com.

[12] http://www.pitchforkmedia.com.

13.2.2.1 Acoustic Features: Timbral, Temporal, and Tonal

Acoustic and musical features used by existing music recommenders include:

- *timbral* features, such as Mel-frequency cepstrum coefficients (MFCCs) [79, 82, 104, 147] and other features related to spectral shape of the signal [17, 32, 76, 92];
- *temporal* and *time-domain* features, characterizing temporal evolution of loudness and timbre [17, 76, 104, 137], rhythmic properties such as beat (tempo) histogram features [17, 55, 76] and onset rate [17, 81], average loudness and dynamics [17, 32];
- *tonal* features, such as harmonic pitch class profiles (chroma) [17, 136, 142] or similar pitch-based features [55, 76, 81], key, scale, chords distribution, and inharmonicity measures [17, 32, 81].

Timbral, temporal, and tonal information address different aspects of music, and can be combined to provide a solid foundation for recommendation algorithms. However, until recently, these different approaches were rarely integrated in academic studies.

Timbral similarity, which compares spectral shapes of the tracks, is probably the most basic and common similarity that can be applied for audio-based music recommendation. Timbre information can be represented as probability distributions of the frame-wise MFCCs, and compared using a number of distance metrics [8, 80, 141]. In particular, Logan [79] considered average, median, and minimum MFCC-based distance from tracks in a target music collection to the preferred tracks and a distance to the summarized MFCC distribution of all preferred tracks. Subjective evaluations of such MFCC-based approaches revealed only average or below-average user satisfaction [17, 82] and suggested their insufficiency compared to approaches with larger feature sets containing a combination of timbral, temporal, and tonal features [17].

Some studies implement wider varieties of acoustic features and include temporal and tonal dimensions of music, which may be complemented with metadata. Pampalk et al. [103, 104] expand timbral similarity based on MFCCs [8] with temporal information that includes fluctuation patterns and derived descriptors of distinctiveness of the fluctuations at specific frequencies and of the overall perceived tempo. Su et al. [137] proposed a music recommender that encodes the temporal evolution of timbral information as time sequences of timbre clusters. The system infers preferred and disliked sequences based on the user's previous track ratings, and matches the feature distribution of recommended tracks to the user's profile.

Celma and Herrera [32] propose an approach based on Euclidean distance, which uses timbre, dynamics, tempo, meter, rhythmic patterns, tonal strength, key, and mode information. This approach is compared to an item-based collaborative filtering distance using listening statistics from *Last.fm*. A large-scale evaluation is conducted, the results of which suggest that the collaborative filtering approach is better able to predict which tracks a user would like, but also produces recommendations which are more familiar to the user. Importantly, this study corroborates that content-based approaches can be effectively incorporated in order to increase novelty of recommendations without a devastating decrease in their quality. Interestingly, average ratings were merely satisfactory: $\approx$3.39 and $\approx$2.87 for collaborative filtering and content-based approaches, respectively, on a 1-to-5 Likert-type liking scale.

Instead of computing similarity between music items and a user profile, some authors propose discriminative models which use audio features to either classify items into *liked* and *disliked* categories or predict user ratings. For example, Grimaldi and Cunningham [55] propose a classification-based approach which uses the tracks rated by a user as *good* and *bad* examples. The authors apply k-nearest neighbors (k-NN) and feature sub-space ensemble classifiers to a set of temporal features derived from beat histograms and tonal features describing harmony. They conclude that the selected audio features are insufficient for the task, except when user preferences are strongly driven by specific genres. Moh et al. [92] propose to classify music into *liked* and *disliked* by using a variety of timbral features, including MFCCs, spectral centroid/rolloff/flux, and zero crossing rate. They evaluate several classification algorithms based on variants of support vector machines (SVMs), as well as a probabilistic Gaussian model to predict user preference.

As an alternative to binary classification, Reed and Lee [116] propose ordinal regression to predict ratings from audio features describing temporal evolution of the MFCCs within each track. Bogdanov [16] investigates the importance of various timbral, temporal, tonal, and semantic features for predicting music preferences. To this end, regression models using these features are built for each particular user in order to predict her ratings.

13.2.2.2 Automatic Semantic Annotation

Currently, collaborative filtering techniques tend to outperform approaches based purely on audio [18, 32, 132]. Audio-based methods are inherently limited in that they cannot (directly) exploit information beyond the pure signal. As a consequence, low-level acoustic descriptors may capture information which has little direct relation to user preference. It is thus desirable to use high-level abstractions or semantic concepts, such as genres, moods, or instrumentation. When these annotations are not provided by human annotators (as described in Sect. 13.2.1), machine learning techniques can be used to predict annotations from audio content.

Bridging the so-called *semantic gap* [6, 33], which arises from the weak linking between human concepts related to musical aspects and the audio-based features, is notoriously difficult. To this end, Barrington et al. [12] propose a semantic music similarity measure which is used for music recommendation. They train Gaussian mixture models (GMMs) of MFCCs for a number of semantic concepts, such as genres, moods, instrumentation, vocals, and rhythm. Thereafter, high-level descriptors are obtained by computing the probabilities of each concept on a frame basis. The resulting semantic annotations of tracks are represented as a distribution over tags, and compared in order to assess similarity. Subsequent work compares this auto-tagging approach to a similarity metric directly derived from

MFCC distributions and finds that the direct MFCC approach is more effective at predicting collaborative filtering similarity between tracks [84]. The authors attribute this finding to the effect of using a fixed set of semantic concepts, which can provide user-interpretable representations, but may prematurely discard useful information for determining similarity. Bogdanov et al. [17] propose a similarity-based recommendation approach grounded on an extensive set of over 60 timbral, temporal, and tonal features together with automatic semantic annotations by genre, mood, instrumentation, and rhythm, created by probabilistic SVMs.

13.3 Contextual Music Recommendation

The topic of context-awareness has gained popularity in recommender systems research in recent years [1] (see Chap. 6 for an extensive review). However, the idea of using context information in computing applications can be traced back to the 1990s. One of the first works in this area defined context as "*information describing where you are, whom you are with, and what resources are nearby*" [127]. In other words, context can be considered as any information that influences the interaction of the user with the system. For instance, in the domain of music recommendation, context can be the situation of the user when listening to recommended tracks (e.g., time, mood, current activity, the presence of other people). Clearly, such information may influence the user's appreciation of music and thus it could be taken into account, in addition to the more conventional knowledge of the user's long-term preferences, when providing recommendations.

Various classifications of contextual information have been proposed in the literature. Adomavicius et al. [1] distinguish between *fully observable*, *partially observable*, and *unobservable* context, where unobservable context may be modeled using latent features that influence the changes in user's short-term preferences [56]. Dey and Abowd [38] suggest distinguishing between the *primary* and *secondary* context. The primary context is defined as the user's location, identity, activity, and time. The authors argue that these four factors are the most important ones when characterizing a user's situation. The secondary context is defined as additional information which can be derived from the primary context factors. For instance, the current weather conditions may be derived from the user's location and time.

In this section, we categorize context information into two general classes—*environment-related* context, which consists of features that can be measured by sensors on the user's mobile device or obtained from external information services e.g., the user's location, current time, weather, temperature, etc., and *user-related* context, which is difficult to measure directly and represents a more high-level information about the user e.g., the user's activity, emotional state, or social environment. Similarly to the relation between primary and secondary context defined by Dey and Abowd [38], environment-related context may be used to derive the user-related context.

13.3.1 Environment-Related Context

A user's environment, such as season, temperature, time of day, noise level, weather conditions, etc., has an influence on the user's state of mind, and therefore indirectly influences her musical preferences. Research has shown that there exists a correlation between characteristics of the listening situation and the preference for music that augments these characteristics [96]. For instance, people tend to prefer different types of music in summer and in winter [109]. Consequently, it may be beneficial to consider environment-related context attributes when recommending music content. Such attributes used in music recommendation research can be classified into the following groups:

- *Location* of the user can be represented by a ZIP code, geographical coordinates, type of landscape (e.g., city, nature), nearby monuments, buildings, landmarks, etc. The surroundings of the user may have a strong impact on her perception and preferences of music. The US music duo Bluebrain is the first band to record a location-aware album.[13] In 2011, the band released two such albums—one dedicated to Washington's park National Mall, and the second dedicated to New York's Central Park. Both albums were released as iPhone apps, with music tracks pre-recorded for specific zones in the parks. As the listener moves through the landscape, the tracks change through smooth transitions, providing a soundtrack to the walk. Despite the large potential of location-aware music services, up to date there has been little research exploring location-related context information in music recommendations.
- *Time* information may refer to the time of day (typically categorized into morning, afternoon, evening, night), or day of week (can be represented by the exact day or can be categorized into working day, weekend). This kind of information is potentially useful since studies have shown that user's music preferences differ depending on the day of the week or moment of the day [60].
- *Weather* information may refer to weather conditions (typically categorized into sunny, overcast, rainy, etc.), to the temperature (e.g., cold, moderate, hot), or to the season. Such information is relevant for music recommendation since the user's music preferences may significantly differ, e.g., in a cold rainy autumn or a hot sunny midsummer [109].
- *Other factors* such as information about the traffic conditions, the noise level, or the amount of ambient light may contribute to the user's state of mind and therefore indirectly influence her music preferences.

One of the first music recommenders to exploit environment-related context was described by Reddy and Mascia [115]. The authors used information about the user's location (represented by a ZIP code), time of day (morning, afternoon, evening, night), day of week, noise level (calm, moderate, chaotic), temperature

[13] http://bluebrainmusic.blogspot.com/.

(frigid, cold, moderate, warm, hot), and weather (rainy, snow, haze, cloudy, sunny, clear). The described system is capable of recommending songs from the user's music library which have to be tagged using a controlled tag vocabulary, where the tags directly represent the values of context attributes. For instance, to recommend a song for a particular location, it has to be tagged with the appropriate ZIP code.

Ankolekar and Sandholm [5] presented a mobile audio application, *Foxtrot*, that allows its users to explicitly assign audio content to a particular location. The authors stressed the importance of the emotional link between music and location. According to the authors, the primary goal of their system is to "*enhance the sense of being in a place*" by creating its emotional atmosphere. *Foxtrot* relies on crowd-sourcing—the users of *Foxtrot* are allowed to assign audio pieces (either a music track or a sound clip) to specific locations (represented by the geographical coordinates of the user's current location), and also specify the visibility range of the audio track—a circular area within which the track is relevant. The system is then able to provide a stream of location-aware audio content to the users.

Braunhofer et al. [24] explored the possibilities to adapt music to the places of interest (POIs) that the user is visiting. This idea is based on the hypothesis that a fitting music track may enhance the sightseeing experience of the user. For instance, during a visit to a Baroque cathedral a user might enjoy hearing a composition by Bach, while the narrow streets in Venice offer a good surrounding to listen to a Vivaldi's concerto. The matching of music and POIs was made by representing both music tracks and POIs with a common set of emotion tags, motivated by music perception research [148]. In a related research, Fernández-Tobías et al. [47] have developed a technique to recommend music content related to POIs using explicit knowledge about musicians and POIs extracted from *DBpedia*[14] [9]. The tag-based [24] and knowledge-based [47] techniques have been combined and evaluated in a web-based user study [68].

Okada et al. [98] describe a mobile music recommender and define context as "*a finite set of sensed conditions collected from a mobile device*", in other words, the authors focus on environment-related context information: ambient noise, location (represented by geographical coordinates), time of day, and day of week. The authors do not provide a detailed technical description of the recommendation algorithm (i.e., how exactly context is used to select music), but rather focus on the architectural design and usability principles of a context-aware mobile music recommender. The authors describe a user study which shows an overall positive evaluation of the system. However, user feedback suggests the need for explanations of the recommendations and more control over the played songs. This leads to an important research question—how to integrate the features of a regular music player and a context-aware recommender.

[14] http://www.dbpedia.org.

13.3.2 User-Related Context

Any contextual information related to the user may be important when recommending music, since music preferences are linked to people's activities, emotions, or social background. Schäfer and Sedlmeier [118] observe different uses of music to serve listeners' needs, such as the ones related to cognitive, emotional, socio-cultural, and physiological functions. The user-related context used in music recommendation research can be classified into the following groups:

- *Activity* information includes an action, typically represented as an element from the set of possible actions (e.g., walking, running, driving), or a numerical attribute defining the user's state (e.g., walking pace or heart rate). This type of context has been shown to have an impact on the user's musical preferences. Foley [52] has shown that people prefer different musical tempo depending on their occupation. North and Hargreaves [97] related personality traits and social lifestyles to music preferences.
- *Emotional state* or mood has a direct influence on the user's music preferences. For example, a user may wish to listen to different types of music when in a sad mood compared to when being happy. Research has shown that music can be used both to moderate the user's emotional condition [72, 118] and to augment the emotions perceived by the listener [96].
- *Social context* information, i.e., the presence of other people, may influence user's music preferences. For instance, people may choose music taking into account the preferences of their companions. Several works have addressed the issue of generating music playlists for groups of users [10, 113]. Mesnage [90] exploited user relations in social networks for music discovery.
- *Cultural context* is closely related to environment-related context (location), however, it defines a more high-level information, e.g., the user's cultural background or belonging to an ethnic group. Koenigstein et al. [71] have exploited the activity of US-based users in peer-to-peer networks to predict the popularity of music tracks in US song charts. Schedl [121] used geo-tagged tweets to extract location-based music listening trends and in turn build a location-aware recommender system.

Compared to the environment-related context, user-related context is difficult to measure directly using mobile sensors or external information services. However, it can be derived to some extent from the environment-related context attributes. For instance, such context attributes as the time of day, ambient noise level, temperature, weather, etc., were used in Bayesian classifiers to predict the user's emotional state [105] or activity [142].

Emotional state of the user is a particularly popular type of context information, which can be exploited to create emotion-based music recommenders, such as *Musicovery*.[15] In addition to adapting music to the user's mood, emotions have been used

[15] http://www.musicovery.com.

to match music with other types of content that can cause an emotional response in users, e.g., text or images [28, 75, 136]. Emotion-based music recommendation is becoming an increasingly popular topic, largely due to advances in automatic music emotion recognition [146].

13.3.3 Incorporating Context Information in Music Recommender Systems

Having described the main types of context information exploited in music recommender systems, we now turn to the major challenge of designing a context-aware recommender—incorporating context information in the recommendation algorithm. Chapter 6 provides a detailed discussion on the paradigms for incorporating context in recommender systems. We therefore refer the reader to the aforementioned chapter for an in-depth discussion on this topic, and here provide only a brief overview of techniques for exploiting context in music recommenders.

Context is known to have an effect on user preferences and information needs [1]. To exploit this information when recommending music, one must establish a degree of relevance between a music track and the contextual information. This information may be obtained on a per user level, e.g., by having users rate music in a particular situation defined by the context attributes, or it can be established globally, by obtaining a relatedness score between a music track and a context attribute. The relevance of particular contextual attributes for music tracks can then be exploited in a recommendation algorithm.

We define four types of approaches to establish a degree of relevance between a music piece and contextual information, as shown in Fig. 13.1:

1. Rating music in context [11, 105] is an extension of the classical collaborative filtering approach. While suffering from the cold-start problem, this is still the state of the art when designing context-aware recommender systems [1].
2. Mapping low-level music features to context attributes [142] is an approach based on machine learning techniques and is closely related to music information retrieval [30] since it involves audio signal analysis. This approach needs training data of music labeled with appropriate context values.
3. Direct labeling of music with context attributes [5, 115] is the most straightforward approach, whose main disadvantage is the high effort required to label music tracks, similarly to rating music in context.
4. Predicting an intermediate context, such as the user's activity [142] or emotional state [24, 105]. This type of approach incorporates the aforementioned techniques—rating in context [105], mapping low-level music features to context [68, 142], or manual labeling of music with context attributes [24].

In summary, context represents an important source of information which can be combined with other sources, such as music content features or user ratings, to provide highly personalized and adaptive music services. Recommender systems

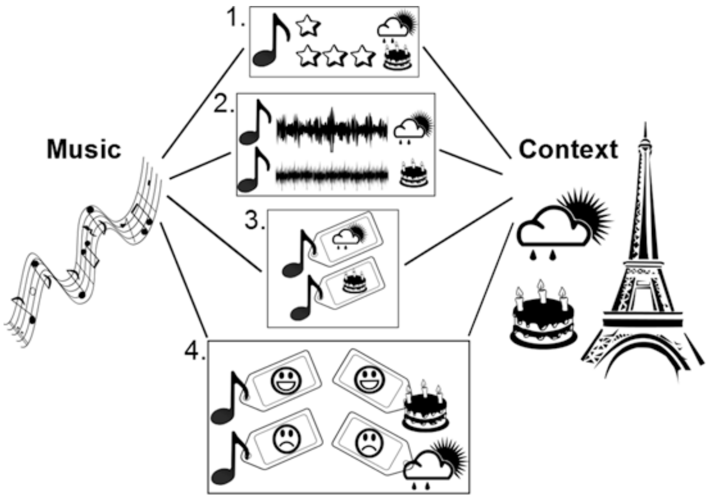

Fig. 13.1 The different types of approaches to establish the relevance of contextual attributes for music

that combine these different sources of information are called hybrid systems. In the next section, we provide a detailed description of hybrid music recommendation and give more details on works that incorporate context information into recommendation algorithms.

13.4 Hybrid Music Recommendation

Since music preference is a complex and multi-faceted concept, it is a logical step to incorporate multiple aspects of musical similarity into recommendation. In the preceding sections, we have discussed different approaches to describe the contents of music and to exploit the context of music consumption. In this section, we discuss hybrid music recommenders, i.e., systems that *"combine two or more recommendation techniques to gain better performance with fewer of the drawbacks of any individual one"* [26]. Before reviewing approaches that integrate different sources, let us briefly reconsider properties of the individual sources used for music recommendation and the entailed advantages and disadvantages.

Like in every other domain, recommendation approaches built upon implicit or explicit user feedback have to deal with the common problems of data sparsity and in particular the cold-start problem. To some extent, this is the same with content-based approaches that rely on external sources for item description. Regardless of whether the content source is editorial metadata, text from the web, or social tags, one or more humans must first create the underlying data. Thus, both of these approaches also exhibit popularity biases in that wider-known items are more likely to have information related to them. By relying on human-crafted data, metadata methods are also susceptible to attacks, vandalism, and manipulation [34, 91]. Even putting aside potential malicious influence, web-based approaches must contend with a great deal of noise in the data. Manual expert annotations, on the other hand, are accurate, but prohibitively costly to scale to large collections [138]. Context features inherently derive from end-users, and are therefore the most difficult to obtain (in academic settings) and typically noisy.

Depending on the type of integration, context-aware recommendation can additionally amplify the problem of data sparsity [1]. Conversely, content-based approaches that extract information directly from the audio signal do not suffer from these problems. Signal-based features provide a static description that can be used for unbiased and time-independent similarity calculation. However, audio content methods have drawbacks as well, such as computational overhead and the requirement of access to the music signal. Moreover, audio content methods are usually outperformed by collaborative filtering and methods that exploit user-generated data [132].

In general, any combination of two or more approaches can be considered a hybrid. For instance, in Sect. 13.2, we described work that combines different types of content-based recommendation. Other approaches combine different aspects of collaborative filtering, such as the *Auralist* framework, which aims at improving user satisfaction by providing diverse and novel recommendations [149]. In the remainder of this section, we focus on work that combines different techniques and information from different sources.

13.4.1 Combining Content with Context Descriptors

To date, there are relatively few methods which combine music content and user context. Schedl [120] presents the *Mobile Music Genius* (MMG) player, which gathers a wide range of user-context attributes during music playback, e.g., time, location, weather, device- and phone-related features (music volume), tasks (running on the device), network, ambient (light, proximity, pressure, noise), motion (accelerometers, orientation), and player-related features (repeat, shuffle, sound effects). MMG then learns relations (using a C4.5 decision tree learner) between these $\sim$100-dimensional feature vectors and metadata (genre, artist, and track are considered), and uses these learned relations to adapt the playlist on the fly when the user's context changes by a certain amount.

Elliott and Tomlinson [45] focus on the particular activities of walking and running. The authors present a system that adapts music to the user's pace by matching the beats per minute of music tracks with the user's steps per minute. Additionally, the system uses implicit feedback by estimating the likelihood of a song being played based on the number of times the user has previously skipped the song. In similar research, de Oliveira and Oliver [99] compare the user's heart rate and steps per minute with music tempo to moderate the intensity of a workout.

Park et al. [105] model a number of context attributes—temperature, humidity, noise, light level, weather, season, and time of day—with Bayesian networks to infer the emotional state of the user: depressing, content, exuberant, or anxious/frantic. The music tracks used in the described system are represented by genre, tempo, and mood attributes. In order to recommend music for the emotional states, users must explicitly express their preferences for each music attribute in every emotional state using a 5-point rating scale. For instance, a user may state that she prefers rock music with a preference rating of 4 in a depressing state, 3 in a content state, and 2 in an exuberant state.

More recently, Wang et al. [142] described a mobile music recommender where the time of day, accelerometer data, and ambient noise are used to predict the user's activity—running, walking, sleeping, working, or shopping. To recommend music for the user's activity context, music tracks had to be labeled with the appropriate activity labels. The authors use a data set of 1200 songs manually labeled with activity values and represented by low-level audio feature vectors for training an auto-tagging algorithm [13].

13.4.2 Combining Collaborative Filtering with Content Descriptors

Collaborative filtering and content descriptors, in particular those extracted from the audio signal, exhibit complementary features. A combination of the two is expected to improve recommendation quality for the following reasons, cf. [26, 27, 31, 41]:

- *Avoiding cold-start problems*: While new items are lacking preference data, audio content analysis and comparison to all existing items can be performed instantly. Thus, when no user feedback is available, a hybrid system could resort to audio similarity for recommendation.
- *Avoiding popularity biases*: Preference data, as well as content metadata, may be focused on popular items only, whereas audio-based information is available uniformly. Including objective content descriptors can remove recommendation biases.
- *Increasing novelty and diversity*: Popularity biases can result in a limited range of recommended items, whereas audio-based approaches are agnostic to whether music is a hit or from the long tail. Therefore, new and lesser known items are more likely to be recommended when both sources are exploited.
- *Combining information on usage with musical knowledge*: Recommendation in the multi-faceted domain of music should benefit from the incorporation of sources reflecting different aspects of music perception.

A straightforward approach to incorporating both preference and content information is to create independent recommenders and combine their outputs using a meta-classifier (*ensemble learning*). Following this direction, Tiemann and Pauws [139] implement an item-based collaborative filtering recommender as well as a content-based recommender that integrates timbre, tempo, genre, mood, and release year features. Both recommenders predict ratings as weighted combinations of the most similar items' ratings. For the final rating prediction, the feature vectors constructed from the individual recommenders' predictions are compared to the output vectors from the learning phase using Euclidean distance and the rating of the most similar vector is predicted. The idea of *fusing outputs* of multiple recommenders is also applied by Lu and Tseng [81], who combine three rankings, namely a ranking according to content similarity based on features extracted from the musical score, a ranking according to user-based collaborative filtering over a data set of user surveys, and an emotion-based ranking in accordance with manual emotion annotations by an expert. In the combination step, a personalization component is introduced. This component reweights the individual rankings according to user feedback gathered in an initial survey in which users specified preference assessments (likes/dislikes) and the underlying reasons (such as preference by tonality, rhythm, etc.) for a sample of tracks.

Instead of fusing multiple outputs in a late stage, preference and content can be *integrated earlier*, for instance to generate a new set of multi-modal features or to adapt similarity measures. The challenge is to combine sources in a manner that avoids the individual drawbacks rather than propagating them. For instance, a simple feature concatenation or unsupervised linear combination can easily preserve the data sparsity problems of preference-based approaches [130].

McFee et al. [84] optimize a content-based similarity metric by learning from a sample of collaborative data. First, a codebook representation of delta-MFCCs is learned to represent songs as a histogram over the derived codewords. Applying metric learning to rank, the resulting feature space is optimized to reflect item similarity according to implicit feedback, i.e., listening histories of users. This allows to find similar items even for novel and unpopular items based on audio content, while maintaining high recommendation accuracy resulting from feedback data.

Van den Oord et al. [100] follow this general direction, but exploit latent space descriptions of both audio features and implicit feedback (song play counts). First, a weighted matrix factorization algorithm [62] is used to learn latent factor representations of users and songs from usage data. Second, log-compressed Mel-spectrograms of randomly sampled 3-second-windows from the songs are presented to a convolutional neural network [61], preserving temporal relations in music to some extent. Here, the latent factor vectors obtained from the weighted matrix factorization step serve as ground truth to train the network. It is shown that this latent factor modeling of audio optimized for latent factor information on usage outperforms traditional MFCC-based vector quantization methods using linear regression or a multi-layer perceptron for latent factor prediction, as well as the metric learning to rank method by McFee et al.

For integrating heterogeneous data into a single, unified, multi-modal similarity space, McFee and Lanckriet [88] propose a multiple kernel learning technique. They demonstrate the applicability of their technique on a music similarity task on the artist level by including five data sources representing different aspects of an artist, namely artist timbre (modeled over all delta-MFCCs extracted from all songs by the artist), auto-tags, social tags, biographical text, and collaborative filtering data. Comparing the unified similarity space with individual similarity spaces (and partial combinations) against a human-annotated ground truth shows that the multiple kernel learning technique outperforms an unweighted combination of individual kernels. It can also be seen that the timbre similarity performs poorly (potentially since it is originally targeting the song level rather that the artist level) and that social tags contribute the most valuable information.

Another group of hybrid music recommenders combines user feedback and content information by means of a *probabilistic framework*. Li et al. [76] propose a probabilistic model in which music tracks are pre-classified into groups by means of both audio content (timbral, temporal, and tonal features) and user ratings. Predictions are made for users considering the Gaussian distribution of user ratings given the probability that a user belongs to a group Yoshii et al. [147] propose a hybrid probabilistic model, in which each music track is represented as a vector of weights of timbres (a "bag-of-timbres"), i.e., as a GMM over MFCCs. Each Gaussian corresponds to a single timbre. The Gaussian components are chosen universally across tracks, being predefined on a certain music collection. Ratings and "bags-of-timbres" are associated with latent variables, conceptually corresponding to genres, and music preferences of a particular listener can be represented in terms of proportions of the genres. A three-way aspect model (a Bayes network) is proposed for this mapping, with the idea that a user stochastically chooses a genre according to her/his preference, and then the genre stochastically "generates" pieces and timbres.

Several approaches follow a *graph-based interpretation* of musical relations to integrate different sources. In the resulting models, the vertices correspond to the songs, and the edge weights correspond to the degree of similarity. Shao et al. [130] build such a model upon a hybrid similarity measure that automatically re-weights a variety of audio descriptors in order to optimally reflect user preference. On the resulting song graph, rating prediction is treated as an iterative propagation of ratings from rated data to unrated data.

Multiple dimensions of similarity can be expressed simultaneously using a *hypergraph*—a generalization in which "hyperedges" can connect arbitrary subsets of vertices. Bu et al. [25] compute a hybrid distance from a hypergraph which contains MFCC-based similarities between tracks, user similarities according to collaborative filtering of listening behavior from *Last.fm*, and similarities on the graph of *Last.fm* users, groups, tags, tracks, albums, and artists, i.e., all possible interactions that can be crawled from *Last.fm*. The proposed approach is compared with user-based collaborative filtering, a content-based timbral approach, and their hybrid combination, on a listening behavior data set. Again, the performance of a timbral approach fell behind the ones working with collaborative filtering, while incorporation of all types of information showed the best results.

McFee and Lanckriet [87] build a hypergraph on a wide range of music descriptors to model and, subsequently, generate playlists by performing random walks on the hypergraph (cf. Sect. 13.5). Hypergraph edges are defined to reflect subsets of songs that are similar in some respect. The different modes of similarity are derived from the *Million Song Dataset* (MSD, cf. Sect. 13.6.3), and include:

- *Collaborative filtering similarity*: connects all songs via an edge that are assigned to the same cluster after k-means clustering for $k = \{16, 64, 256\}$ on a low-rank factorization of the user-song matrix;
- *Low-level acoustic similarity*: connects all songs assigned to the same cluster after k-means clustering for $k = \{16, 64, 256\}$ on audio features;
- *Musical era*: connects songs from the same year or same decade;
- *Familiarity*: connects songs with the same level of popularity (expressed in the categories low, medium, and high);
- *Lyrics*: connects songs assigned to the same topic derived via latent Dirichlet allocation (LDA) [15];
- *Social tags*: connects songs assigned to the same *Last.fm* tag;
- *Pairwise feature conjunctions*: creates a category for any pairwise intersection of the described features and connects songs that match both;
- *Uniform shuffle*: an edge connecting all songs in case no other transition is possible.

The weights of the hypergraph are learned using the *AotM-2011* data set, a collection of over 100,000 unique playlists crawled from *Art of the Mix*[16] (cf. Sect. 13.6.6). In addition to playlist information, this data set also contains a timestamp and a categorical label, such as *romantic* or *reggae*, for each playlist. Experiments on a global hypergraph with weights learned from all playlists and on category-specific hypergraphs trained only on the corresponding subsets of playlists show that performance can be improved when treating specific categories individually ("playlist dialects"). In terms of features, again, social tags have the most significant impact on the overall model, however audio features are more relevant for specific categories such as *hip hop*, *jazz*, and *blues*, whereas lyrics features receive stronger weights for categories like *folk* and *narrative*.

The category labels of the *AotM-2011* data set exhibit further interesting aspects. While most labels refer to genre categories, some refer to a usage scenario or the *user-related context* of a playlist. We discuss these aspects next.

13.4.3 Combining Collaborative Filtering with Context Descriptors

In this section, we review hybrid approaches that incorporate models of user preference and user-related context. As discussed in the previous section, the method proposed by McFee and Lanckriet [87] uses different recommenders for

[16] http://www.artofthemix.org.

different categories, some of which refer to a user's activity (*road trip*, *sleep*), emotional state (*depression*), or social situation (*break up*). The results indicate that the influence of different aspects of musical content can vary dramatically, depending on contextual factors.

The approach by Baltrunas et al. [11] to recommend driving music takes advantage of ratings specifically assigned to each contextual condition (*context-aware collaborative filtering* [1], cf. Sect. 13.3.2). For incorporating environmental (such as *traffic* and *weather*) and user-related factors (such as *mood* and *sleepiness*) into rating prediction, they extend a matrix factorization approach to collaborative filtering by introducing one additional parameter for each pair-wise combination of contextual condition and musical genre to the model. The parameters of the model are then learned using stochastic gradient descent. It is shown that mean absolute error (MAE) decreases when incorporating contextual factors.

Typically, the user-related context is not explicitly available in the observed data. In such cases, hidden context can be modeled by latent factor techniques. Hariri et al. [56] propose a method to apply sequential pattern mining on an LDA model of playlists from *Art of the Mix*, in which songs are represented by social tags from *Last.fm*. While the LDA topics should reflect the contextual factors affecting listening preference—e.g., mood or social setting—sequential pattern mining should capture changes in context over time. Predictions of the listener's current context then provide the additional information to build a context-aware music recommender. Hariri et al. show that the LDA-based context-aware recommender significantly outperforms a simple metadata-based recommendation approach.

Taking a similar approach, Zheleva et al. [150] also apply LDA to a set of listening histories extracted from usage logs of the *Zune Social* platform[17] over a period of 14 weeks. They compare two approaches. The first, called *taste model*, is a direct application of the LDA method developed for text collections and thus refers to overall factors of listening preference. The second, called *session model*, incorporates additional information about listening sessions and aims at capturing latent factors related to mood in a more consistent listening context. Evaluation of the approaches is carried out on the genre level, i.e., instead of predicting individual songs or specific artists, a recommendation consists of a distribution of genres. Furthermore, the discovered taste topics are compared to genres within the two-leveled *Zune Social* genre taxonomy. Evaluation indicates that the context-aware session model is more effective than the time-agnostic taste model. Yang et al. [145] investigate "local preferences," i.e., temporal aspects on a smaller and more consistent time scale. These preferences reflect changes in listening behavior that are strongly influenced by the listening context and occurring events rather than caused by a gradual change in general taste.

The impact of the *temporal context* is not limited to listening sessions. Temporal information is also helpful for modeling long-term patterns in listening behavior

[17] http://zune.net; now *Xbox Music*.

and song life cycles. Dror et al. [43] show that matrix factorization models for music rating prediction can successfully incorporate additional information such as temporal dynamics in listening behavior, temporal dynamics in item histories, and multi-level taxonomy information like genre. Aizenberg et al. [3] apply collaborative filtering methods to the playlists of radio stations associated with the web radio station directory *ShoutCast*.[18] Their goals include prediction of existing radio station programs, as well as predicting the programs of new radio stations. To this end, they model latent factor station affinities as well as temporal effects. We discuss the specifics of sequential recommendation in greater detail in the next section.

13.5 Automatic Playlist Generation

One of the key distinguishing features of music, as compared to other item domains such as books or movies, is that recommendations are often consumed in rapid succession during a listening session. Rather than selecting each song individually, a sequence of songs—a *playlist*—can be automatically generated, and the user would consume the sequence much as if it was a traditional radio broadcast. Automatic playlist generation thus forms a critical component of personalized streaming radio services and portable music devices.

Because the user does not explicitly select or provide feedback for each song in a playlist, the modeling assumptions and evaluation criteria can differ from those of traditional recommender systems (Sect. 8). In this section, we survey evaluation methodologies and algorithmic approaches for automatic playlist generation.

13.5.1 Parallel and Serial Consumption

In most typical recommendation models, the user is first provided with a set of candidate items from which to choose, for example, a page of movie recommendations. The user may then inspect each candidate item before making a selection: in effect, the user can access the candidate recommendations in *parallel*. The selection process may be assisted by presenting the user with a brief summary of each item, such as a star rating, plot synopsis, or capsule review. This approach works well for browsing scenarios in which the user is actively engaged and selecting each item individually.

Unlike browsing a collection, playlist consumption is an inherently *serial* process: only one song is consumed at a time, and the user does not select from a set of alternatives. Typical playlist consumption interfaces mimic conventional

[18] http://www.shoutcast.com.

radio or personal music devices, potentially augmented with a limited set of familiar controls, such as *skip*, *stop*, or *pause*. Because the mode of consumption differs from that of browsing, the semantics and availability of user feedback differ as well. The semantics of explicit per-song feedback are straightforward, but events may be rare due to user disengagement (passive consumption) or fatigue due to consuming a large number of songs in rapid succession.

Implicit feedback can be somewhat more problematic. If a song plays to completion, it may be interpreted as implicit positive feedback, but it is also possible that the user has become disengaged—e.g., by reducing the volume or wandering away—and there is often no way to infer this behavior directly. Negative feedback, on the other hand, must derive from an explicit user action, such as clicking a "stop" or "skip" button [64, 104]. However, as noted by Bosteels et al. [23], great care must be taken when inferring intent from a user's intent action: the user may in fact dislike the recommended song, or she may simply not wish to hear it at that moment due to otherwise obscure contextual factors.

13.5.2 Playlist Evaluation

Sequential playlist consumption differs from traditional recommender system and information retrieval settings, and consequently, several methods have been proposed to evaluate playlist generation algorithms. Because the choice of evaluation criteria influences algorithm design, we first provide a survey of evaluation techniques. At a high level, these techniques fall into four categories which we survey in this section: user studies, semantic cohesion, partial playlist prediction, and generative likelihood.

13.5.2.1 User Studies

Early approaches to evaluating automatic playlist generation systems relied upon user studies. For example, Pauws and Eggen [106] conducted a study in which users were asked to provide a *seed song* in response to a pre-selected contextual query (e.g., *lively music*), which was then used to seed a playlist generation algorithm. Each user then rated the resulting playlist on a scale of 1–10. Later studies followed this general approach by soliciting users for ratings of playlist consistency [111] and similarity to the seed song [20]. Alternatively, Barrington et al. [12] conducted a survey in which users were provided with a seed song and playlists generated by two competing systems, and asked for relative preference of one playlist or the other.

While user studies provide high-quality information, they are notoriously difficult to reproduce, and they do not provide a viable means of automatically evaluating algorithms in a laboratory setting. User evaluation is also difficult to scale to large collections, as the search space of playlists grows exponentially with the number of songs in the collection.

13.5.2.2 Semantic Cohesion

A commonly used alternative to user-centric evaluation is to measure some notion of cohesion over the songs within a playlist. This general strategy is usually applied to song-level metadata, for example, by counting the fraction of songs in the playlist by the same artist [78], or measuring the entropy of the playlist's genre distribution [42, 69, 111]. In cohesion-based playlist evaluation, the metadata in question is obscured from the playlist generation algorithm.

The main drawback of cohesion-based evaluation is that it is essentially user-agnostic, so one cannot directly conclude that an algorithm which produces cohesive playlists will also produce satisfactory recommendations to users. On the contrary, a study conducted by Slaney and White [133] provides evidence that users prefer some degree of diversity in playlists.

13.5.2.3 Partial Playlist Prediction

Rather than evaluate each automatically generated playlist, some authors have evaluated their algorithm's ability to predict the hidden songs in pre-existing playlists from a partial observation. Platt et al. [110] gather a collection of user-generated playlists over a fixed library of songs. For each playlist in the collection, the algorithm is given as input a partial observation of the constituent songs, and as output, produces a ranking over the remaining songs in the library. The algorithm is then evaluated according to the position within the predicted ranking of the remaining songs in the playlist.

Maillet et al. [83] conduct a similar experiment, in which playlists are collected by mining the playback logs of terrestrial broadcast radio stations. Their evaluation methodology is similar to that of Platt et al., except that the partial observations are restricted to immediately preceding song(s), rather than arbitrary partial observations.

Partial prediction evaluation is similar to ranking-based evaluations commonly used in general implicit-feedback collaborative filtering problems [63, 117]. One key distinction, however, is that associations are measured between playlists and songs, not users and songs. Because playlists tend to be much shorter than a user's full listening history, the associations tend to be sparse when compared to a full collaborative filter (see Fig. 13.2). As noted by Platt et al., the sparsity of observations, coupled with the general lack of strong negative feedback, tends to result in an overly pessimistic evaluation [110].

13.5.2.4 Generative Likelihood

The final approach to playlist evaluation is borrowed from the statistical natural language processing community. McFee and Lanckriet [86] argue that because many practical playlist generation algorithms are stochastic, they induce probability

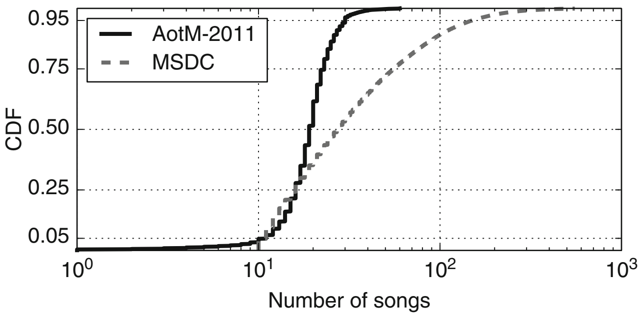

Fig. 13.2 The empirical cumulative distribution function (CDF) of the number of songs in user-generated playlists (*solid line*), and in a user's listening history (*dashed line*). Playlists were gathered from the *Art of the Mix* (AotM-2011) corpus, which includes approximately 10^5 unique playlists [87]. Listening histories were gathered from the *Million Song Dataset Challenge* (MSDC) training set, which contains listening histories for approximately 10^6 users [85]. Ninety-five percent of playlists contain 30 or fewer songs, indicating a high degree of sparsity in the observations. Note that these sets do not span the same user base

distributions over playlists. The induced distribution can thus be interpreted as model of the data (sample playlists), and evaluated in a similar fashion to a natural language model. Concretely, the algorithm is scored according to the likelihood of a test collection of playlists under its corresponding distribution.

In practice, the generative likelihood approach requires a large test corpus of sample playlists. Test corpora can be formed from user-constructed playlists [86, 87], or broadcast or streaming radio logs [93, 94]. However, when evaluating on historical data, rather than intentionally constructed playlists, one must be aware that the data itself may have been generated by an automated process.

The generative likelihood approach only applies to algorithms for which a sample playlist's likelihood can be computed. While this includes broad families of algorithms, such as Markov processes [86], it rules out direct comparisons to deterministic algorithms and black-box methods, e.g., existing streaming radio services. However, the generative likelihood approach does provide a consistent evaluation framework, and a meaningful objective function for designing and optimizing playlist generation algorithms.

13.5.3 Playlist Generation Algorithms

A wide range of algorithmic techniques have been proposed for automatic playlist generation. Most techniques fall into one of three categories, which we survey here. *Constraint satisfaction* methods attempt to construct a playlist which satisfies some user-specified search criteria. *Similarity heuristic* methods build playlists by finding songs which are in some way similar to a query or seed song. Finally, *machine learning* approaches can be used to optimize model parameters over a training set of example playlists.

13.5.3.1 Constraint Satisfaction

Early research into automatic playlist generation algorithms primarily focused on combinatorial methods. Common formulations of the playlist generation problem required the user to encode her query in the form of a set of constraints which must be satisfied by the generated playlist [4, 7, 102, 107]. Usually, constraints would be applied to metadata associated with each song (e.g., genre or year of release), or audio content analysis (e.g., track duration or tempo). Pauws et al. [107] identify several types of constraints, including *unary* (e.g., "each song must be of the *jazz* genre"), *binary* (e.g., "adjacent songs must have similar loudness"), and *global* (e.g., "total duration less than 60 minutes").

Research on constraint-based playlist generation has tailed off in recent years due to several practical limitations. First, constraint satisfaction problems tend to be computationally intractable even for relatively small personal collections, making them unattractive for large-scale applications [7]. Second, because constraint satisfaction is a feasibility problem, and not an optimization problem, there is no explicit notion of *preference* between two satisfactory playlists. Consequently, it may take multiple interactive refinements before the user is satisfied with the recommendations [106]. Finally, constraint generation can be a difficult task for users who may lack the technical sophistication to clearly express their preferences. However, it should be noted that constraint satisfaction forms a necessary component of automatic playlist generators for broadcast radio and streaming services, which may be required by law to conform to certain regulations [48, Sect. 2.7.3].

13.5.3.2 Similarity Heuristics

As an alternative to the query-by-constraint formulations described above, several researchers have proposed methods which allow the user to formulate a query in the form of one or more *seed songs*. Playlists may then be composed by selecting songs which are in some way similar to the seed.

The underlying notion of similarity between songs ultimately determines the song selection, and many different approaches have been proposed in the literature. Most commonly, song similarity is determined by acoustic content features, such as MFCCs, rhythmic descriptors, or automatic semantic annotations [12, 20, 42, 51, 78, 104, 112]. Alternative methods of computing similarity between songs include metadata (e.g., genre or mood) [110], proximity of artists in a social network [49], or textual similarity extracted from web documents [69].

Given one or more seed songs and a song-level similarity function, several methods have been proposed to generate a playlist. In the simplest form, the playlist is constructed by ranking songs by similarity to the seed(s) [12, 78, 110]. More sophisticated approaches construct a graph over songs, and use path-finding algorithms to navigate between seeds, such as shortest path [51], network flow [49], and traveling salesman [69, 112].

13.5.3.3 Machine Learning Approaches

In each of the similarity-based examples above, the notion of similarity between songs is fixed a priori, and is not informed by user activity. However, most recent techniques use some form of machine learning to optimize model parameters from a training set of playlists.

The algorithm proposed by Ragno et al. [114] generates playlists by performing random walks on an undirected graph where edge weights are determined by co-occurrence of songs within training playlists. By relying strictly on playlist co-occurrence, the algorithm is implicitly constrained to only reproduce previously observed sequences. Other authors proposed methods which incorporate tag-based similarity [23], latent topic assignment sequences [56], or combine popularity with artist-level co-occurrence [22] to allow the algorithm to generalize and produce novel sequences.

The above methods use co-occurrence frequency counts to inform song selection, but they are not explicitly optimized for playlist prediction. Maillet et al. [83] propose a method to train a classifier to predict from acoustic features whether an ordered pair of songs form a bigram in observed playlists. By keeping the first song fixed, the classifier's output can be used to induce a ranking over the remaining songs in the library, from which the next song is selected. The proposed method also incorporates direct user feedback by using a weighted tag cloud to reorder the candidate selections. Because the method uses a discriminative classifier, the authors synthesized "negative" training example bigrams by random sampling.

Recently, generative modeling has emerged as a versatile framework for developing playlist generation algorithms. In this view, playlists are generated by sampling sequences from a probability distribution whose parameters are fit to a training sample. This approach lends itself well to generative likelihood evaluation (Sect. 13.5.2.4), as the training and testing criteria match exactly. Existing models in the literature exhibit a range of scale and complexity, including latent topic models [150], low-dimensional song embedding [93], co-embedding of songs and users [94], Markov chain mixtures [86], and cross-modal feature integration [87].

13.6 Data Sets and Evaluation

In this section, we give an overview of frequently used data sets and prominent evaluation campaigns in MIR and music recommendation. In cases where data sets were specifically created for the purpose of running an evaluation campaign we discuss them together.

A comparative overview of data sets is given in Tables 13.1 and 13.2. The former lists statistics of the data sets and the type(s) of editorial metadata included, while the latter details the kind of data that is provided. Note that the statistics in Table 13.1 only indicate figures of the data sets that are publicly available for the individual types of items. The last column "Ratings/Evts." refers to explicit (ratings) or implicit

Table 13.1 Statistics of public data sets for music recommendation research

Data set/items	Songs	Albums	Artists	Users	Ratings/evts.
Yahoo! Music [44]	—— 624,961 in total ——			1,000,990	262,810,175
MSD [14]	1,000,000			1,019,318	48,373,586
Last.fm—360K [31]			186,642	359,347	
Last.fm—1K [31]			107,528	992	19,150,868
MusicMicro [121]	71,410		19,529	136,866	594,306
MMTD [57]	133,968		25,060	215,375	1,086,808
AotM-2011 [87]	98,359		17,332	16,204	859,449

Table 13.2 Features of public data sets for music recommendation research

Data set	Feedback type	Audio files	Item content	User context
Yahoo! Music [44]	Ratings	✗	✗	✗
MSD [14]	Listening events, tags	✗	✓	✗
Last.fm—360K [31]	listening events	✗	✓	✗
Last.fm—1K [31]	Listening events	✗	✓	✓
MusicMicro [121]	Listening events	✗	✓	✓
MMTD [57]	Listening events	✗	✓	✓
AotM-2011 [87]	Playlists	✗	✓	Partial

(listening events) preference indications.[19] In Table 13.2, the column "Feedback type" refers to the kind of user-item-relationship that is addressed (e.g., ratings or listening events), whereas "Item content" indicates the presence or absence of content descriptors (e.g., metadata or audio features). The last column "User context" shows whether contextual data of the user or the listening event is provided (e.g., location or time).[20] Note that the absence of audio files in all data sets (see Table 13.1) would render audio content-based approaches impossible. However, some data sets (e.g., MSD) come with precomputed audio features, such as those provided by *The Echo Nest*.[21] If extracting features directly from the audio file is desired, an alternative solution is to download 30-second-snippets frequently available for preview in major online music stores and compute features on these. Such previews are also offered by *7digital*[22] via their Media Delivery API.[23]

In the following, we give a short introduction to the evaluation of music recommendation techniques in general. Hereafter, we present major evaluation

[19]In AotM-2011, this figure refers to the sum of the length of all playlists, where length is measured as the number of songs.

[20]For AotM-2011 this is partially the case, as not all playlist categories refer to contextual factors.

[21]http://the.echonest.com.

[22]http://www.7digital.com.

[23]http://developer.7digital.com/resources/api-docs.

Table 13.3 Some references to works that make use of the discussed data sets

Data set	References
Yahoo! Music [44]	[53, 73]
MSD [14]	[40, 65, 66, 128]
Last.fm—1K/360K [31]	[39, 144]
MusicMicro [121]	[119, 124]
MMTD [57]	[46, 50, 95, 125]
AotM-2011 [87]	[21, 86]

campaigns and data sets explicitly addressing the task of music recommendation.[24] To give the reader some hints on the usage of each data set, Table 13.3 provides references to corresponding work.

13.6.1 Evaluation Methodologies

In the recommender systems community, evaluation is often conducted by measuring the error of predicted ratings (e.g., root-mean-square error, RMSE). Due to the historical shortage of publicly available rating data for music, evaluation of music recommendation approaches has been carried out for a long time using genre as proxy and modeling a genre prediction task. Given the genre of the seed item(s) and that of the recommended item(s), typical IR performance measures are used (e.g., precision and recall). Using genre as proxy for music preferences, however, can be considered inherently incomplete because listeners might have driving factors for preference other than genres (e.g., happy music with vocals). It further neglects the perceived quality of recommendations, their actual usefulness for the listener [129], and the user's satisfaction [89, 122]—aspects which can only be assessed by asking real users.

Although the number of user studies has increased [143], conducting such studies on real-world commercial music collections remains time-consuming, expensive, and impractical, particularly for academic researchers. Consequently, relatively few studies measuring aspects related to user satisfaction have been published. The study by Celma and Herrera [32] may serve as an example of a proper subjective evaluation experiment, carried out on a larger scale. This study was conducted on 288 participants, each of which provided liking (enjoyment of the recommended music) and familiarity ratings for 19 tracks recommended by three approaches in a blind evaluation. The resulting large total number of evaluated tracks served as a solid basis for statistical testing. Bogdanov [16] proposes to use four subjective measures addressing different aspects of user preference and satisfaction to assess

[24]There exist many more music benchmarking activities which are oriented towards retrieval or annotation, e.g., *MIREX* (http://www.music-ir.org/mirex/wiki) or *MusiClef* (http://www.cp.jku.at/datasets/musiclef).

the quality of recommendations: (1) liking; (2) familiarity with the recommended tracks; (3) listening intention, i.e., readiness to listen to the same track again in the future; and (4) "give-me-more," indicating a request for or rejection of more music that is similar to the recommended track.

13.6.2 Yahoo! Music Dataset and KDD Cup 2011

In 2011, the *KDD Cup*[25] [44] featured a music recommendation task using music ratings data gathered on a large scale and provided by *Yahoo!*.[26] The corresponding data set is simply known as the *Yahoo! Music* data set and currently represents the largest music recommendation data set, including 262,810,175 ratings of 624,961 music items by 1,000,990 users, and spanning the time period from 1999 to 2010. User ratings are given partly on a standard 5-point scale, and partly on a 0–100 scale. Different levels of granularity are covered by the ratings: tracks, albums, artists, and genres. A characteristic of the data set is its high sparsity (99.96 %), even in light of the typically sparse nature of other ratings data sets (for instance, 98.82 % for the *Netflix* set) [44]. This high sparsity renders recommendation tasks particularly challenging.

There were two objectives in *KDD Cup 2011*, which were addressed on separate tracks. The first track was a traditional recommendation task: predict unknown music ratings based on given explicit ratings. The best algorithm achieved an RMSE of 0.84, when assuming a 5-point-scale for rating. It was capable of explaining 59.3 % of the rating variance. The second task aimed at distinguishing *loved* songs from songs never rated. In particular, participants were required to predict three songs for each user in the test set. To this end, the test set contained six songs for each user: three of which the user rated high, three of which the user never rated. As performance measure an error rate was used, corresponding to the fraction of songs wrongly predicted as loved ones. For this second track, a smaller data set was released, roughly 250,000 users, 300,000 items, and 60,000,000 ratings. The best performing algorithm achieved an error rate of 2.47 % [44].

The *KDD Cup 2011* received a lot of attention and had more than 2000 participants. However, it was also the subject of some controversy within the MIR community (see http://musicmachinery.com/2011/02/22/is-the-kdd-cup-really-music-recommendation).

The main criticism stemmed from the total anonymization and absence of any descriptive metadata. Both users and items are represented only by opaque numerical identifiers that do not relate to any semantic entity, such as user name or editorial music metadata. The task was therefore frequently considered as applying collaborative filtering techniques to a huge data set, rather than addressing the

[25] http://www.sigkdd.org/kdd2011/kddcup.shtml.

[26] http://music.yahoo.com.

particularities of music recommendation. The data set and the challenge effectively ignore music domain knowledge, and as a result, prohibit the application of content-based approaches. Nevertheless, the *Yahoo! Music* data set still represents one of the largest collections of user ratings on music items.

13.6.3 Million Song Dataset (MSD) and MSD Challenge 2012

Acknowledging the fact that human music perception is not only influenced by aspects encoded in the audio signal, the proponents of the *Million Song Dataset*[27] (MSD) [14] brought together a wealth of descriptors and information on one million contemporary popular music pieces. As of the time of writing, MSD contains content-based descriptors (e.g., estimates of key, tempo, loudness) and editorial metadata (e.g., artist, title, release year) from *The Echo Nest*, links to *MusicBrainz* and *7digital*, collaborative tags and similarity information from *Last.fm*, term vector representation of song lyrics from *musiXmatch*,[28] user playcount information (called "taste profile") again from *The Echo Nest* (covering almost 50 million `<user, song, playcount>` triples for about one million users), and information about cover songs from *Second Hand Songs*.[29]

Even though it has been criticized by some MIR researchers, foremost for (1) lack of actual audio material and (2) non-transparency of how the content descriptors were obtained, MSD certainly marked a cornerstone of publicly available music-related data sets in terms of size and data variety. In this vein, the proponents encourage MIR research that scales to commercial sizes of music collections. As for criticism (1), although it is true that MSD does not come with the actual digital song files due to copyright reasons, 30-second-snippets can be downloaded easily via links to *7digital*. Criticism (2) originates from the fact that the content descriptors are provided out of the box by *The Echo Nest*, which does not reveal details on how they were computed. Users of MSD however are also free and encouraged to compute their own audio-based features from the *7digital* snippets.

In order to provide an open evaluation contest for music recommendation algorithms that can use a wide variety of data sources, the *MSD Challenge*[30] [85] was organized in 2012. In contrast to *KDD Cup 2011*, which was highly obscured in terms of available data, the MSD Challenge put strong emphasis on allowing for a wide variety of approaches (for instance, including web crawling, audio analysis, collaborative filtering, or use of metadata).

Given full listening histories of one million users and half of the listening histories for another 110,000 test users, the task was to predict the missing hidden

[27] http://labrosa.ee.columbia.edu/millionsong.

[28] http://www.musixmatch.com.

[29] http://www.secondhandsongs.com.

[30] http://labrosa.ee.columbia.edu/millionsong/challenge.

listening events for the test users.[31] Mean average precision (MAP) computed on the top 500 recommendations for each listener was used as main performance measure. The winning algorithm achieved a MAP of 17.91 % using a neighborhood method [2]. The proponents of the MSD Challenge further provided some simple reference implementations that recommended songs only based on their popularity, achieving MAP scores between 2.1 % and 2.3 %.

As noted above, several publicly available data sets are strongly tied to their respective evaluation campaigns. This does not mean that they were only used in the corresponding campaigns though; quite the contrary is true. However, there exist a few collections that were proposed independently of benchmarking initiatives. A selection is presented in the following.

13.6.4 Last.fm Dataset: 360K/1K Users

In his book "Music Recommendation and Discovery" [31], Celma proposes the *Last.fm Dataset—360K users* and the *Last.fm Dataset—1K users*.[32] The former contains listening information about almost 360,000 users, but only includes artists they most frequently listened to. The latter provides full listening histories of nearly 1000 users, up to May 2009. While the *360K* set contains `<user, artist, playcount>` triples, the *1K* set further contains information on which songs were played at which time, thus representing the data as `<user, timestamp, artist, song>` quadruples. Both data sets contain user-specific information, including gender, age, country, and date of registering at *Last.fm*. The data has been gathered via the *Last.fm* API.

13.6.5 MusicMicro and Million Musical Tweets Dataset (MMTD)

The importance of temporal and spatial information has been highlighted in context-aware recommender systems in general [1], but also particularly in music recommendation [36, 124]. Until 2013, however, no music-related data set providing both types of information in high granularity was publicly available. Although Celma's data set contains timestamps of listening events, location is only given on the user level. Based on music listening information extracted from microblogs, two data sets were proposed in 2013: *MusicMicro* [121] and the *Million Musical Tweets*

[31] http://www.kaggle.com/c/msdchallenge.

[32] http://ocelma.net/MusicRecommendationDataset.

Dataset (MMTD) [57].[33] The *MusicMicro* set contains about 600,000 listening events by almost 137,000 distinct users and 21,000 artists. MMTD encompasses 1,087,000 listening events by 215,000 *Twitter* users, referring to 25,000 different artists. The latter data set can be regarded as an extension of the former. Temporal information is provided as month and weekday, and spatial information is given as numerical longitude and latitude values, as well as respective countries and cities. In addition, MMTD further includes identifiers linking to *MusicBrainz*, *7digital*, and *Amazon*.

13.6.6 AotM-2011

The *AotM-2011* data set[34] [87] contains playlists crawled from *Art of the Mix*,[35] a portal to share music playlists of any kind. The playlists span the time period from January 1998 to June 2011. The data set contains 101,343 unique playlists, which contain a total of 859,449 events (i.e., song-playlist pairs). Each playlist has had its songs matched to the *Million Song Dataset*, resulting in a total of 98,359 matching tracks. Furthermore, a timestamp of the playlist's upload is provided. Some of the playlists are further annotated with activities. In addition, metadata (name and date of joining the *Art of the Mix* site) is supplied for each user.

13.7 Conclusions and Challenges

In this chapter, we have given a brief overview of the state of the art in music recommender systems. We described the distinguishing characteristics of music recommendation in comparison to other domains, and surveyed content-based, context-aware, hybrid, and serial recommendation methods. We further reviewed common data sets, evaluation strategies and campaigns, and outlined their limitations.

From a practical point of view, there is no single best solution to music recommendation, in terms of features or algorithms. However, a trend towards hybrid approaches, in particular incorporating context-aware aspects is evident.

The overarching challenge for music recommendation research is comprehensive access to large data sets, including not only user ratings, but also contextual information and audio content. From the researcher's perspective, this further motivates the need for efficient and scalable methods which can be applied to large collections. Unfortunately, publicly available data sets with full access to audio are

[33] http://www.cp.jku.at/datasets/musicmicro and http://www.cp.jku.at/datasets/MMTD, resp.

[34] http://bmcfee.github.io/data/aotm2011.html.

[35] http://www.artofthemix.org.

rare and usually small, and therefore not amenable to recommender evaluation. On the other hand, companies that are in possession of large collections are not eager to share their data, be it due to business reasons, user privacy concerns, or legal constraints (i.e., copyright).

Beyond the issues of data access, there is also a need for better understanding how different kinds of data (e.g., semantic descriptions, audio content, or contextual factors) relate to and influence human music perception. Although many of the studies described in this chapter have evaluated some of these effects in isolation or on small data sets, there is still a relative lack of large-scale, comprehensive user studies for music recommendation. Whenever possible, evaluations should be carried out with real users, instead of optimizing for traces of preference that do not reveal any background information or intent [122]. Moreover, even with a better understanding of how individual factors influence music perception, it is still unclear how to best integrate all available sources when developing hybrid recommenders.

Regarding the state of the art in context-aware music recommendation, we note that most systems presented in Sects. 13.3 and 13.4 are research prototypes. While certain music players allow specifying the user's mood or activity as a query, to our knowledge, no fully automated context-aware music recommenders have been released to the public. The research on context-awareness in the music domain is still in its early stages and more work is needed to address such important research topics as understanding the relations between contextual conditions and music [97, 109], explaining context-aware recommendations to users, and determining the right level of user control over the recommendations [98].

If the research community manages to address these challenges and transcend current limitations in music recommendation, many more exciting applications can be expected in the future. These may include music players that "understand" the user's information or entertainment need at any point in time and provide corresponding recommendations, or applications that target specific usage scenarios such as group recommendations.

References

1. Adomavicius, G., Mobasher, B., Ricci, F., Tuzhilin, A.: Context-aware recommender systems. AI Magazine **32**, 67–80 (2011)
2. Aiolli, F.: A preliminary study on a recommender system for the million songs dataset challenge. Preference Learning: Problems and Applications in AI p. 1 (2012)
3. Aizenberg, N., Koren, Y., Somekh, O.: Build your own music recommender by modeling internet radio streams. In: Proceedings of the 21st International Conference on World Wide Web, pp. 1–10. ACM, New York, NY, USA (2012)
4. Alghoniemy, M., Tewfik, A.: Music Playlist Generation Based on Global and Transitional Constraints. In: IEEE Transactions on Multimedia (2003)
5. Ankolekar, A., Sandholm, T.: Foxtrot: a soundtrack for where you are. In: Proceedings of Interacting with Sound Workshop: Exploring Context-Aware, Local and Social Audio Applications, pp. 26–31 (2011)

6. Aucouturier, J.J.: Sounds like teen spirit: Computational insights into the grounding of everyday musical terms. In: J. Minett, W. Wang (eds.) Language, Evolution and the Brain, Frontiers in Linguistics, pp. 35–64. Taipei: Academia Sinica Press (2009)
7. Aucouturier, J.J., Pachet, F.: Scaling Up Music Playlist Generation. In: Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2002), pp. 105–108. Lausanne, Switzerland (2002)
8. Aucouturier, J.J., Pachet, F., Sandler, M.: "The way it sounds": timbre models for analysis and retrieval of music signals. IEEE Trans. on Multimedia **7**(6), 1028–1035 (2005).
9. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: DBpedia: A nucleus for a web of open data. In: ISWC'08, pp. 722–735 (2008)
10. Baccigalupo, C.G.: Poolcasting: an intelligent technique to customise musical programmes for their audience. Ph.D. thesis, Universitat Autònoma de Barcelona (2009)
11. Baltrunas, L., Kaminskas, M., Ludwig, B., Moling, O., Ricci, F., Lüke, K.H., Schwaiger, R.: InCarMusic: Context-Aware Music Recommendations in a Car. In: International Conference on Electronic Commerce and Web Technologies (EC-Web). Toulouse, France (2011)
12. Barrington, L., Oda, R., Lanckriet, G.: Smarter than genius? Human evaluation of music recommender systems. In: Proceedings of the 10th International Conference on Music Information Retrieval (ISMIR'09), pp. 357–362 (2009)
13. Bertin-Mahieux, T., Eck, D., Maillet, F., Lamere, P.: Autotagger: A model for predicting social tags from acoustic features on large music databases. Journal of New Music Research **37**(2), 115–135 (2008).
14. Bertin-Mahieux, T., Ellis, D.P., Whitman, B., Lamere, P.: The million song dataset. In: Proceedings of the 12th International Society for Music Information Retrieval Conference, pp. 591–596. Miami, USA (2011)
15. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet Allocation. Machine Learning Research **3**, 993–1022 (2003)
16. Bogdanov, D.: From music similarity to music recommendation: Computational approaches based in audio features and metadata. Ph.D. thesis, Universitat Pompeu Fabra, Barcelona, Spain (2013)
17. Bogdanov, D., Haro, M., Fuhrmann, F., Xambó, A., Gómez, E., Herrera, P.: Semantic audio content-based music recommendation and visualization based on user preference examples. Information Processing & Management **49**(1), 13–33 (2013).
18. Bogdanov, D., Herrera, P.: How much metadata do we need in music recommendation? a subjective evaluation using preference sets. In: Int. Society for Music Information Retrieval Conf. (ISMIR'11), pp. 97–102 (2011)
19. Bogdanov, D., Herrera, P.: Taking advantage of editorial metadata to recommend music. In: Int. Symp. on Computer Music Modeling and Retrieval (CMMR'12) (2012).
20. Bogdanov, D., Serrà, J., Wack, N., Herrera, P., Serra, X.: Unifying Low-Level and High-Level Music Similarity Measures . IEEE Transactions on Multimedia **13**(4), 687–701 (2011)
21. Boland, D., Murray-Smith, R.: Information-theoretic Measures of Music Listening Behaviour. In: Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR 2014). Taipei, Taiwan (2014)
22. Bonnin, G., Jannach, D.: Evaluating the quality of playlists based on hand-crafted samples. In: 14th International Society for Music Information Retrieval Conference, ISMIR (2013)
23. Bosteels, K., Pampalk, E., Kerre, E.: Evaluating and analysing dynamic playlist generation heuristics using radio logs and fuzzy set theory. In: Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR (2009)
24. Braunhofer, M., Kaminskas, M., Ricci, F.: Location-aware music recommendation. International Journal of Multimedia Information Retrieval **2**(1), 31–44 (2013).
25. Bu, J., Tan, S., Chen, C., Wang, C., Wu, H., Zhang, L., He, X.: Music recommendation by unified hypergraph: combining social media information and music content. In: ACM Int. Conf. on Multimedia (MM'10), pp. 391–400 (2010).
26. Burke, R.: Hybrid recommender systems: Survey and experiments. User Modeling and User-Adapted Interaction **12**(4), 331–370 (2002)

27. Burke, R.: Hybrid web recommender systems. In: The adaptive web, pp. 377–408 (2007)
28. Cai, R., Zhang, C., Wang, C., Zhang, L., Ma, W.Y.: Musicsense: contextual music recommendation using emotional allocation modeling. In: MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia, pp. 553–556. ACM, New York, NY, USA (2007)
29. Casey, M.A., Veltkamp, R., Goto, M., Leman, M., Rhodes, C., Slaney, M.: Content-based music information retrieval: Current directions and future challenges. Proceedings of the IEEE **96**(4), 668–696 (2008)
30. Casey, M.A., Veltkamp, R., Goto, M., Leman, M., Rhodes, C., Slaney, M.: Content-Based Music Information Retrieval: Current Directions and Future Challenges. Proceedings of the IEEE **96**, 668–696 (2008)
31. Celma, Ò.: Music Recommendation and Discovery – The Long Tail, Long Fail, and Long Play in the Digital Music Space. Springer, Berlin, Germany (2010)
32. Celma, O., Herrera, P.: A new approach to evaluating novel recommendations. In: ACM Conf. on Recommender Systems (RecSys'08), pp. 179–186 (2008)
33. Celma, O., Herrera, P., Serra, X.: Bridging the music semantic gap. In: ESWC 2006 Workshop on Mastering the Gap: From Information Extraction to Semantic Representation (2006). Available online: http://mtg.upf.edu/node/874
34. Celma, O., Lamere, P.: Music recommendation and discovery revisited. In: Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011), pp. 7–8. ACM, New York, NY, USA (2011)
35. Celma, O., Ramírez, M., Herrera, P.: FOAFing the music: A music recommendation system based on RSS feeds and user preferences. In: Int. Conf. on Music Information Retrieval (ISMIR'05) (2005)
36. Cunningham, S., Caulder, S., Grout, V.: Saturday Night or Fever? Context-Aware Music Playlists. In: Proceedings of the 3rd International Audio Mostly Conference of Sound in Motion (2008)
37. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. Journal of the American Society for Information Science **41**, 391–407 (1990)
38. Dey, A., Abowd, G.: Towards a better understanding of context and context-awareness. In: CHI 2000 workshop on the what, who, where, when, and how of context-awareness, vol. 4, pp. 1–6 (2000)
39. Diaz-Aviles, E., Georgescu, M., Nejdl, W.: Swarming to Rank for Recommender Systems. In: Proceedings of the 6th ACM Conference on Recommender Systems. Dublin, Ireland (2012)
40. Dieleman, S., Brakel, P., Schrauwen, B.: Audio-based music classification with a pretrained convolutional network. In: Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011). Miami, FL, USA (2011)
41. Donaldson, J.: A Hybrid Social-Acoustic Recommendation System for Popular Music. In: Proceedings of the ACM Recommender Systems (RecSys 2007). Minneapolis, MN, USA (2007)
42. Dopler, M., Schedl, M., Pohle, T., Knees, P.: Accessing Music Collections via Representative Cluster Prototypes in a Hierarchical Organization Scheme. In: Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR'08). Philadelphia, PA, USA (2008)
43. Dror, G., Koenigstein, N., Koren, Y.: Yahoo! Music Recommendations: Modeling Music Ratings with Temporal Dynamics and Item Taxonomy. In: Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011). Chicago, USA (2011)
44. Dror, G., Koenigstein, N., Koren, Y., Weimer, M.: The Yahoo! Music Dataset and KDD-Cup'11. Journal of Machine Learning Research: Proceedings of KDD-Cup 2011 competition **18**, 3–18 (2012)
45. Elliott, G.T., Tomlinson, B.: Personalsoundtrack: context-aware playlists that adapt to user pace. In: CHI '06: CHI '06 extended abstracts on Human factors in computing systems, pp. 736–741. ACM, New York, NY, USA (2006)

46. Farrahi, K., Schedl, M., Vall, A., Hauger, D., Tkalčič, M.: Impact of Listening Behavior on Music Recommendation. In: Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR 2014). Taipei, Taiwan (2014)
47. Fernández-Tobías, I., Cantador, I., Kaminskas, M., Ricci, F.: Knowledge-based music retrieval for places of interest. In: Proceedings of the 2nd International Workshop on Music Information Retrieval with User-Centered and Multimodal Strategies (MIRUM), pp. 19–24 (2012)
48. Fields, B.: Contextualize your listening: the playlist as recommendation engine. Ph.D. thesis, Department of Computing Goldsmiths, University of London (2011)
49. Fields, B., Rhodes, C., Casey, M., Jacobson, K.: Social playlists and bottleneck measurements: Exploiting musician social graphs using content-based dissimilarity and pairwise maximum flow values. In: ISMIR, pp. 559–564 (2008)
50. Figueiredo, F., Almeida, J.M., Matsubara, Y., Ribeiro, B., Faloutsos, C.: Revisit behavior in social media: The phoenix-r model and discoveries. In: Proceedings of the 7th European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2014). Nancy, France (2014)
51. Flexer, A., Schnitzer, D., Gasser, M., Widmer, G.: Playlist generation using start and end songs. In: ISMIR, pp. 173–178 (2008)
52. Foley Jr, J.: The occupational conditioning of preferential auditory tempo: a contribution toward an empirical theory of aesthetics. The Journal of Social Psychology **12**(1), 121–129 (1940)
53. Goel, S., Broder, A., Gabrilovich, E., Pang, B.: Anatomy of the Long Tail: Ordinary People with Extraordinary Tastes. In: Proceedings of the 3rd ACM International Conference on Web Search and Data Mining (WSDM 2010). New York, USA (2010)
54. Green, S.J., Lamere, P., Alexander, J., Maillet, F., Kirk, S., Holt, J., Bourque, J., Mak, X.W.: Generating transparent, steerable recommendations from textual descriptions of items. In: ACM Conf. on Recommender Systems (RecSys'09), pp. 281–284 (2009)
55. Grimaldi, M., Cunningham, P.: Experimenting with music taste prediction by user profiling. In: ACM SIGMM Int. Workshop on Multimedia Information Retrieval (MIR'04), pp. 173–180 (2004)
56. Hariri, N., Mobasher, B., Burke, R.: Context-aware music recommendation based on latent-topic sequential patterns. In: Proceedings of the sixth ACM conference on Recommender systems, pp. 131–138. ACM (2012)
57. Hauger, D., Schedl, M., Košir, A., Tkalčič, M.: The Million Musical Tweets Dataset: What Can We Learn From Microblogs. In: Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR 2013). Curitiba, Brazil (2013)
58. Haupt, J.: Last.fm: People-powered online radio. Music Reference Services Quarterly **12**(1), 23–24 (2009).
59. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. on Information Systems **22**(1), 5–53 (2004).
60. Herrera, P., Resa, Z., Sordo, M.: Rocking around the clock eight days a week: an exploration of temporal patterns of music listening. In: ACM Conf. on Recommender Systems. Workshop on Music Recommendation and Discovery (Womrad 2010), pp. 7–10 (2010)
61. Hinton, G., Deng, L., Yu, D., Dahl, G.E., rahman Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T.N.,, Kingsbury, B.: Deep neural networks for acoustic modeling in speech recognition. IEEE Signal Processing Magazine **29**(6), 82–97 (2012)
62. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, pp. 263–272. IEEE Computer Society, Washington, DC, USA (2008)
63. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: Eighth IEEE International Conference on Data Mining, ICDM, pp. 263–272 (2008)
64. Hu, Y., Ogihara, M.: Nextone player: A music recommendation system based on user behaviour. In: Int. Society for Music Information Retrieval Conf. (ISMIR'11) (2011)

65. Hu, Y., Ogihara, M.: Genre classification for million song dataset using confidence-based classifiers combination. In: Proceedings of the 35th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR) (2012)
66. Humphrey, E.J., Nieto, O., Bello, J.P.: Data driven and discriminative projections for large-scale cover song identification. In: Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR 2013). Curitiba, Brazil (2013)
67. Jones, N., Pu, P.: User technology adoption issues in recommender systems. In: Networking and Electronic Commerce Research Conf. (2007)
68. Kaminskas, M., Ricci, F., Schedl, M.: Location-aware Music Recommendation Using Auto-Tagging and Hybrid Matching. In: Proceedings of the 7th ACM Conference on Recommender Systems (RecSys 2013). Hong Kong, China (2013)
69. Knees, P., Pohle, T., Schedl, M., Widmer, G.: Combining Audio-based Similarity with Web-based Data to Accelerate Automatic Music Playlist Generation. In: Proceedings of the 8th ACM SIGMM International Workshop on Multimedia Information Retrieval (MIR'06). Santa Barbara, CA, USA (2006)
70. Knees, P., Schedl, M.: A survey of music similarity and recommendation from music context data. ACM Transactions on Multimedia Computing, Communications and Applications **10**(1), 2:1–2:21 (2013)
71. Koenigstein, N., Shavitt, Y., Zilberman, N.: Predicting billboard success using data-mining in p2p networks. In: Multimedia, 2009. ISM'09. 11th IEEE International Symposium on, pp. 465–470. IEEE (2009)
72. Konecni, V.: Social interaction and musical preference. The psychology of music pp. 497–516 (1982)
73. Koren, Y., Sill, J.: OrdRec: an Ordinal Model for Predicting Personalized Item Rating Distributions. In: Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011). Chicago, USA (2011)
74. Levy, M., Sandler, M.: Learning latent semantic models for music from social tags. Journal of New Music Research **37**(2), 137–150 (2008)
75. Li, C.T., Shan, M.K.: Emotion-based impressionism slideshow with automatic music accompaniment. In: MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia, pp. 839–842. ACM Press, New York, NY, USA (2007)
76. Li, Q., Myaeng, S.H., Kim, B.M.: A probabilistic music recommender considering user opinions and audio features. Information Processing & Management **43**(2), 473–487 (2007)
77. Lim, D., Mcfee, B., Lanckriet, G.R.: Robust structural metric learning. In: S. Dasgupta, D. Mcallester (eds.) Proceedings of the 30th International Conference on Machine Learning (ICML-13), vol. 28, pp. 615–623. JMLR Workshop and Conference Proceedings (2013).
78. Logan, B.: Content-based Playlist Generation: Exploratory Experiments. In: Proceedings of the 3rd International Symposium on Music Information Retrieval (ISMIR 2002), pp. 295–296. Paris, France (2002)
79. Logan, B.: Music recommendation from song sets. In: Int. Conf. on Music Information Retrieval (ISMIR'04), pp. 425–428 (2004)
80. Logan, B., Salomon, A.: A music similarity function based on signal analysis. In: IEEE Int. Conf. on Multimedia and Expo (ICME'01), p. 190 (2001).
81. Lu, C.C., Tseng, V.S.: A novel method for personalized music recommendation. Expert Systems with Applications **36**(6), 10,035–10,044 (2009)
82. Magno, T., Sable, C.: A comparison of signal-based music recommendation to genre labels, collaborative filtering, musicological analysis, human recommendation, and random baseline. In: Int. Conf. on Music Information Retrieval (ISMIR'08), pp. 161–166 (2008)
83. Maillet, F., Eck, D., Desjardins, G., Lamere, P.: Steerable playlist generation by learning song similarity from radio station playlists. In: Proceedings of the 10th International Conference on Music Information Retrieval (2009)
84. McFee, B., Barrington, L., Lanckriet, G.: Learning content similarity for music recommendation. IEEE Trans. on Audio, Speech, and Language Processing **20**(8), 2207–2218 (2012).

85. McFee, B., Bertin-Mahieux, T., Ellis, D., Lanckriet, G.: The million song dataset challenge. In: Proc. of the 4th International Workshop on Advances in Music Information Research (AdMIRe) (2012)
86. McFee, B., Lanckriet, G.: The Natural Language of Playlists. In: Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR). Miami, FL, USA (2011)
87. McFee, B., Lanckriet, G.: Hypergraph Models of Playlist Dialects. In: Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR 2012). Porto, Portugal (2012)
88. McFee, B., Lanckriet, G.R.: Learning multi-modal similarity. Journal of Machine Learning Research **12**, 491–523 (2011)
89. McNee, S., Riedl, J., Konstan, J.: Being accurate is not enough: how accuracy metrics have hurt recommender systems. In: CHI'06 extended abstracts on Human Factors in Computing Systems, p. 1101 (2006)
90. Mesnage, C.S.: Social shuffle: music discovery with tag navigation and social diffusion. Ph.D. thesis, Università della Svizzera italiana, Lugano, Switzerland (2011)
91. Mobasher, B., Burke, R., Bhaumik, R., Williams, C.: Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. ACM Transactions on Internet Technology **7**(4) (2007)
92. Moh, Y., Orbanz, P., Buhmann, J.M.: Music preference learning with partial information. In: IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'08), pp. 2021–2024 (2008)
93. Moore, J.L., Chen, S., Joachims, T., Turnbull, D.: Learning to embed songs and tags for playlist prediction. In: 13th International Society for Music Information Retrieval Conference, ISMIR, pp. 349–354 (2012)
94. Moore, J.L., Chen, S., Joachims, T., Turnbull, D.: Taste over time: The temporal dynamics of user preferences. In: 14th International Society for Music Information Retrieval Conference, ISMIR (2013)
95. Moore, J.L., Joachims, T., Turnbull, D.: Taste Space Versus the World: An Embedding Analysis of Listening Habits and Geography. In: Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR 2014). Taipei, Taiwan (2014)
96. North, A., Hargreaves, D.: Situational influences on reported musical preference. Psychomusicology: Music, Mind and Brain **15**(1–2), 30–45 (1996)
97. North, A., Hargreaves, D.: The social and applied psychology of music. Cambridge Univ Press (2008)
98. Okada, K., Karlsson, B.F., Sardinha, L., Noleto, T.: Contextplayer: Learning contextual music preferences for situational recommendations. In: SIGGRAPH Asia 2013 Symposium on Mobile Graphics and Interactive Applications, p. 6. ACM (2013)
99. de Oliveira, R., Oliver, N.: Triplebeat: enhancing exercise performance with persuasion. In: Proceedings of the 10th international conference on Human computer interaction with mobile devices and services, pp. 255–264. ACM (2008)
100. van den Oord, A., Dieleman, S., Schrauwen, B.: Deep content-based music recommendation. In: C. Burges, L. Bottou, M. Welling, Z. Ghahramani, K. Weinberger (eds.) Advances in Neural Information Processing Systems 26, vol. 26, p. 9. Neural Information Processing Systems Foundation (NIPS), Lake Tahoe, NV, USA (2013)
101. Pachet, F.: Knowledge management and musical metadata. Idea Group (2005)
102. Pachet, F., Roy, P., Cazaly, D.: A combinatorial approach to content-based music selection. Multimedia, IEEE **7**(1), 44–51 (2000)
103. Pampalk, E., Flexer, A., Widmer, G.: Improvements of audio-based music similarity and genre classification. In: Int. Conf. on Music Information Retrieval (ISMIR'05), pp. 628–633 (2005)
104. Pampalk, E., Pohle, T., Widmer, G.: Dynamic playlist generation based on skipping behavior. In: Int. Conf. on Music Information Retrieval (ISMIR'05), pp. 634–637 (2005)
105. Park, H.S., Yoo, J.O., Cho, S.B.: A context-aware music recommendation system using fuzzy bayesian networks with utility theory. In: FSKD 2006. LNCS (LNAI), pp. 970–979. Springer (2006)

106. Pauws, S., Eggen, B.: PATS: Realization and user evaluation of an automatic playlist generator. In: Proceedings of the 2nd International Symposium on Music Information Retrieval, ISMIR (2002)
107. Pauws, S., Verhaegh, W., Vossen, M.: Fast generation of optimal music playlists using local search. In: ISMIR, pp. 138–143. Citeseer (2006)
108. Pazzani, M., Billsus, D.: Learning and revising user profiles: The identification of interesting web sites. Machine Learning **27**(3), 313–331 (1997).
109. Pettijohn, T., Williams, G., Carter, T.: Music for the seasons: Seasonal music preferences in college students. Current Psychology pp. 1–18 (2010)
110. Platt, J.C., Burges, C.J.C., Swenson, S., Weare, C., Zheng, A.: Learning a gaussian process prior for automatically generating music playlists. In: Advances in Neural Information Processing Systems. MIT Press (2002)
111. Pohle, T., Knees, P., Schedl, M., Pampalk, E., Widmer, G.: "Reinventing the Wheel": A Novel Approach to Music Player Interfaces. IEEE Transactions on Multimedia **9**, 567–575 (2007)
112. Pohle, T., Pampalk, E., Widmer, G.: Generating Similarity-based Playlists Using Traveling Salesman Algorithms. In: Proceedings of the 8th International Conference on Digital Audio Effects (DAFx-05), pp. 220–225. Madrid, Spain (2005)
113. Popescu, G., Pu, P.: Probabilistic game theoretic algorithms for group recommender systems. In: Proceedings of the 2nd Workshop on Music Recommendation and Discovery (WOMRAD) (2011)
114. Ragno, R., Burges, C.J., Herley, C.: Inferring similarity between music objects with application to playlist generation. In: Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval, pp. 73–80. ACM (2005)
115. Reddy, S., Mascia, J.: Lifetrak: music in tune with your life. In: Proceedings of the 1st ACM International Workshop on Human-centered Multimedia (HCM), pp. 25–34 (2006)
116. Reed, J., Lee, C.: Preference music ratings prediction using tokenization and minimum classification error training. IEEE Trans. on Audio, Speech, and Language Processing **19**(8), 2294–2303 (2011).
117. Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: Bpr: Bayesian personalized ranking from implicit feedback. In: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, pp. 452–461. AUAI Press (2009)
118. Schäfer, T., Sedlmeier, P.: From the functions of music to music preference. Psychology of Music **37**(3), 279–300 (2009)
119. Schedl, M.: #nowplaying Madonna: A Large-Scale Evaluation on Estimating Similarities Between Music Artists and Between Movies from Microblogs . Information Retrieval (2012)
120. Schedl, M.: Ameliorating Music Recommendation: Integrating Music Content, Music Context, and User Context for Improved Music Retrieval and Recommendation. In: Proceedings of the 11th International Conference on Advances in Mobile Computing & Multimedia (MoMM 2013). Vienna, Austria (2013)
121. Schedl, M.: Leveraging Microblogs for Spatiotemporal Music Information Retrieval. In: Proceedings of the 35th European Conference on Information Retrieval (ECIR 2013). Moscow, Russia (2013)
122. Schedl, M., Flexer, A., Urbano, J.: The neglected user in music information retrieval research. Journal of Intelligent Information Systems **41**, 523–539 (2013).
123. Schedl, M., Pohle, T., Knees, P., Widmer, G.: Exploring the Music Similarity Space on the Web. ACM Transactions on Information Systems **29**(3), 1–24 (2011)
124. Schedl, M., Schnitzer, D.: Hybrid Retrieval Approaches to Geospatial Music Recommendation. In: Proceedings of the 36th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR). Dublin, Ireland (2013)
125. Schedl, M., Vall, A., Farrahi, K.: User Geospatial Context for Music Recommendation in Microblogs. In: Proceedings of the 37th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR). Gold Coast, Australia (2014)
126. Schedl, M., Widmer, G., Knees, P., Pohle, T.: A music information system automatically generated via web content mining techniques. Information Processing & Management **47**(3), 426–439 (2011).

127. Schilit, B., Adams, N., Want, R.: Context-aware computing applications. In: Proceedings of the Workshop on Mobile Computing Systems and Applications, pp. 85–90. IEEE Computer Society (1994)
128. Schindler, A., Mayer, R., Rauber, A.: Facilitating comprehensive benchmarking experiments on the million song dataset. In: Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR 2012). Porto, Portugal (2012)
129. Shani, G., Gunawardana, A.: Evaluating recommender systems. Recommender Systems Handbook pp. 257–298 (2009).
130. Shao, B., Wang, D., Li, T., Ogihara, M.: Music recommendation based on acoustic features and user access patterns. IEEE Transactions on Audio, Speech, and Language Processing **17**(8), 1602–1611 (2009)
131. Shardanand, U., Maes, P.: Social information filtering: algorithms for automating "word of mouth". In: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 210–217 (1995)
132. Slaney, M.: Web-scale multimedia analysis: Does content matter? IEEE Multimedia **18**(2), 12–15 (2011)
133. Slaney, M., White, W.: Measuring playlist diversity for recommendation systems. In: 1st ACM workshop on Audio and music computing multimedia, AMCMM '06, pp. 77–82. ACM, New York, NY, USA (2006)
134. Slaney, M., White, W.: Similarity Based on Rating Data. In: Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR 2007). Vienna, Austria (2007)
135. Sordo, M., Celma, O., Blech, M., Guaus, E.: The quest for musical genres: Do the experts and the wisdom of crowds agree? In: Int. Conf. of Music Information Retrieval (ISMIR'08), pp. 255–260 (2008)
136. Stupar, A., Michel, S.: Picasso - to sing, you must close your eyes and draw. In: 34th ACM SIGIR Conf. on Research and development in Information, pp. 715–724 (2011)
137. Su, J.H., Yeh, H.H., Tseng, V.S.: A novel music recommender by discovering preferable perceptual-patterns from music pieces. In: ACM Symp. on Applied Computing (SAC'10), pp. 1924–1928 (2010)
138. Szymanski, G.: Pandora, or, a never-ending box of musical delights. Music Reference Services Quarterly **12**(1), 21–22 (2009).
139. Tiemann, M., Pauws, S.: Towards ensemble learning for hybrid music recommendation. In: ACM Conf. on Recommender Systems (RecSys'07), pp. 177–178 (2007)
140. Turnbull, D.R., Barrington, L., Lanckriet, G., Yazdani, M.: Combining audio content and social context for semantic music discovery. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, pp. 387–394. ACM (2009)
141. Tzanetakis, G., Cook, P.: Musical genre classification of audio signals. IEEE Trans. on Speech and Audio Processing **10**(5), 293–302 (2002)
142. Wang, X., Rosenblum, D., Wang, Y.: Context-aware mobile music recommendation for daily activities. In: Proceedings of the 20th ACM international conference on Multimedia, pp. 99–108. ACM (2012)
143. Weigl, D., Guastavino, C.: User Studies in the Music Information Retrieval Literature. In: Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011). Miami, FL, USA (2011)
144. Weston, J., Wang, C., Weiss, R., Berenzweig, A.: Latent Collaborative Retrieval. In: Proceedings of the 29th International Conference on Machine Learning (ICML). Edinburgh, Scotland (2012)
145. Yang, D., Chen, T., Zhang, W., Lu, Q., Yu, Y.: Local implicit feedback mining for music recommendation. In: Proceedings of the 6th ACM Conference on Recommender Systems, pp. 91–98. ACM (2012)
146. Yang, Y.H., Chen, H.H.: Machine recognition of music emotion: A review. ACM Trans. Intell. Syst. Technol. **3**(3), 40:1–40:30 (2012).

147. Yoshii, K., Goto, M., Komatani, K., Ogata, T., Okuno, H.G.: An efficient hybrid music recommender system using an incrementally trainable probabilistic generative model. IEEE Trans. on Audio, Speech, and Language Processing **16**(2), 435–447 (2008).
148. Zentner, M., Grandjean, D., Scherer, K.R.: Emotions evoked by the sound of music: Characterization, classification, and measurement. Emotion **8**(4), 494–521 (2008)
149. Zhang, Y.C., Séaghdha, D.O., Quercia, D., Jambor, T.: Auralist: Introducing serendipity into music recommendation. In: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, WSDM '12, pp. 13–22. ACM, New York, NY, USA (2012)
150. Zheleva, E., Guiver, J., Mendes Rodrigues, E., Milić-Frayling, N.: Statistical models of music-listening sessions in social media. In: Proceedings of the 19th international conference on World wide web, WWW '10, pp. 1019–1028. ACM, New York, NY, USA (2010)