Music Recommendation Systems: Techniques, Use Cases, and Challenges

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

1 Introduction

In the past decade, we have experienced a drastic change in the way how people search for and consume music. The rise of digital music distribution, followed by the spiraling success of music streaming services such as those offered by Spotify,[1] Pandora,[2] Apple,[3] Amazon,[4] YouTube,[5] and Deezer,[6] has led to the ubiquitous

[1] https://www.spotify.com.

[2] https://www.pandora.com.

[3] https://www.apple.com/apple-music.

[4] https://music.amazon.com.

[5] https://www.youtube.com.

[6] https://www.deezer.com.

M. Schedl (✉)
Johannes Kepler University Linz, Institute of Computational Perception, Multimedia Mining and Search Group, Linz, Austria

LIT AI Lab, Human centered AI Group, Linz, Austria
e-mail: markus.schedl@jku.at

P. Knees
TU Wien, Faculty of Informatics, Institute of Information Systems Engineering, Vienna, Austria
e-mail: peter.knees@tuwien.ac.at

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

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

F. Ricci et al. (eds.), *Recommender Systems Handbook*,
https://doi.org/10.1007/978-1-0716-2197-4_24

availability of music. While the catalogs of these big players are mostly geared towards Western music, in recent years, worldwide, many platforms focusing on domestic markets and music emerged, including Taiwanese KKBOX,[7] Korean Melon,[8] Nigerian Boomplay Music,[9] and Brazilian Superplayer.[10]

As a result, music listeners are suddenly faced with an unprecedented scale of readily available musical content, which can easily become burdensome. Addressing this issue, music recommender systems (MRS) provide support to users accessing large collections of music items and additional music-related content. Music items that are most commonly recommended include artists, albums, tracks, and playlists. Moreover, integrating additional music-related content into their catalogs has become more and more important for streaming providers, to offer their users a unique selling proposition. Such additional content include lyrics, music video clips and animated video backgrounds, album cover images, and information about concert venues.

This chapter gives an introduction to music recommender systems research. In the remainder of this section, we next discuss the unique characteristics of the music recommendation domain (Sect. 1.1), as compared to other content domains, such as videos or books. Then, we define the scope and structure of the subsequent sections (Sect. 1.2).

1.1 Characteristics of the Music Recommendation Domain

There exist several distinguishing characteristics of the music domain that differentiates MRS from other kinds of recommender systems. We summarize the major ones in the following. For a more detailed treatment, we refer the reader to [158, 161].

Duration of consumption: The amount of time required for a user to consume a single media item strongly differs between different categories of items: an image (typically a few seconds), a song (typically a few minutes), a movie (typically one to a few hours), a book (typically days or weeks). Since music ranges at the lower end of the duration scale, the time it takes for a user to form opinions on a music item can be much shorter than in most other domains. As a result, music items may be considered more disposable.

Catalog size: Typical commercial music catalogs contain tens of millions of songs or other musical pieces while catalogs of movies and TV series are several magnitudes smaller. The scalability of commercially used MRS algorithms is, therefore, a more important requirement in the music domain than in other domains.

[7] https://www.kkbox.com.

[8] https://www.melon.com.

[9] https://www.boomplay.com.

[10] https://www.superplayer.fm.

Different representations and abstraction levels: Another distinguishing property is that music recommendations can be made at different item abstraction levels and modalities. While movie recommender systems typically suggest individual items of one specific category (e.g., movies or series) to the user, MRS may recommend music items of various representations and modalities (most commonly, the audio of a song, but also music videos or even digital score sheets offered by providers such as OKTAV[11] or Chordify[12]). Music recommendations can also be effected at different levels of granularity (e.g., at the level of artist, album, or song). Furthermore, non-standard recommendation tasks exist in the music domain, such as recommending radio stations or concert venues.

Repeated consumption: A single music item is often consumed repeatedly by a user, even multiple times in a row. In contrast, other media items are commonly consumed at most a few times. This implies that a user might not only tolerate, but actually appreciate recommendations of already known items.

Sequential consumption: Unlike movies or books, songs are frequently consumed in sequence, e.g., in a listening session or playlist. As a result, sequence-aware recommendation problems [143] such as automatic playlist continuation or next-track recommendation play a crucial role in MRS research. Because of the unique constraints and modeling assumptions of serial consumption, also the evaluation criteria substantially differ from the more standard techniques found in the recommender systems literature [71].

Passive consumption: Unlike most other media content, music is often consumed passively, in the background, which can affect the quality of preference indications. Especially when relying on implicit feedback to infer music preferences of users, the situation where a listener is not paying attention to the music (and therefore does not skip a disliked song) might be misinterpreted as a positive feedback.

Importance of content: In traditional recommendation domains such as movie recommendation, collaborative filtering (CF) techniques have been predominantly used and refined over the years, not least thanks to initiatives such as the *Netflix Prize*.[13] In contrast, research on music recommendation has emerged to a large extent from the fields of audio signal processing and music information retrieval (MIR), and is still strongly connected to these areas. This is one of the reasons why content-based recommendation approaches, such as content-based filtering (CBF), are more important in the music domain than in other domains. Such approaches aim at extracting semantic information from or about music at different representation levels (e.g., the audio signal, artist or song name, album cover, lyrics, album reviews, or score sheet), and subsequently leverage similarities computed on these semantic music descriptors, between items and user profiles, to effect recommendations.[14]

[11] https://www.oktav.com.

[12] https://www.chordify.net.

[13] https://www.netflixprize.com.

[14] 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)

Another reason for the importance of content-based approaches in MRS is the fact that explicit rating data is relatively rare in this domain, and even when available, tends to be sparser than in other domains [44]. Therefore, research in music recommendation techniques tend to rely more upon content descriptions of items than techniques in other domains.

1.2 Scope and Structure of the Chapter

In this chapter, we first categorize in Sect. 2 music recommendation tasks into three major types of use cases. Section 3 subsequently explains the major categories of MRS from a technical perspective, including content-based filtering, sequential recommendation, and recent psychology-inspired approaches. Section 4 is devoted to a discussion of challenges that are faced in MRS research and practice, and of approaches that address these challenges. Finally, in Sect. 5 we conclude by summarizing the main recent trends and open challenges in MRS.

Please note that this chapter substantially differs from the previous version that was published in the second edition of the Recommender Systems Handbook [159]. While the previous version was generally structured according to different techniques and types of music recommender systems, in the version at hand, we take a more user-centric perspective, by organizing our discussion with respect to current use cases and challenges.

2 Types of Use Cases

Research on and development of MRS has evolved significantly over the last decade, owed to changes in the typical use cases of MRS. We can categorize these use cases broadly into basic music recommendation (Sect. 2.1), lean-in (Sect. 2.3), and lean-back experiences (Sect. 2.2).

We refer to the most traditional use case as *basic music recommendation*, which aims at providing recommendations to support users in browsing a music collection. Technically, corresponding tasks are common to other domains, and include predicting a user's explicit rating (rating prediction task) or predicting whether a given user will listen to a particular song (predicting item consumption behavior). Requiring a higher degree of attention and engagement, use cases pertaining to *lean-in* exploration refer to supporting users in searching particular music based on a semantic query that expresses a user intent, e.g., finding music

approaches that operate directly on audio signals and (metadata) approaches that derive item descriptors from external sources, e.g., web documents [90]. In recommender systems research, as in the remainder of this chapter, both types of approaches are described as "content-based".

that fits a certain activity or affective state. In contrast, by providing a *lean-back* experience to the user no specific user task is addressed rather than indulging, for instance, an endless music listening session.

2.1 Basic Music Recommendation

A typical function of a recommender system is to assist in actively browsing the catalog of items through item-to-item recommendations. For an MRS, this implies providing *lists of relevant artists, albums, and tracks*, when a user browses item pages of a music shop or a streaming service. Commonly, such recommendations rely on similarity inferred from the consumption patterns of the users, and they are presented to a user in the form of a list of "people who played that also played this" items.

Another basic functionality is to generate *personalized recommendation lists* on the platform's landing page to engage a user in a session even without their active navigation of the content in the first place. Such recommendations are generated based on the user's previous behavior on the platform, which is a core research task in the recommender systems community. At the same time, it is the topic of lots of user interface (UI) and user experience (UX) design decisions in the industry, often out of the scope of academic research. For example, the system interface may provide contextual "shelves", grouping recommendations by a particular reason, or time span of user activity (e.g., recommendations based on global user profile versus recent user activity).

Figures 1 and 2 demonstrate both types of basic approaches, on the example of Soundcloud[15] and Last.fm.[16] In both cases, such *basic music recommendation* systems deal with artist, album, or track recommendations using the information about previous user interactions and their feedback for the items in the music catalog.

2.1.1 Interaction and Feedback Data

Music services can gather explicit user feedback, including rating provided by a user for artist, album, or track items (e.g., using 1–10 or 1–5 rating scales), or binary Likes, Loves, or Favorites reactions, as well as the information about items purchased or saved to the user's library. Therefore, a common task is predicting those explicit user ratings and reactions for items in the system's music catalog, which is useful to estimate relevance and generate ranked recommendation lists. Historically, user ratings have been associated with online music shops (e.g.,

[15] https://www.soundcloud.com.

[16] https://www.last.fm.

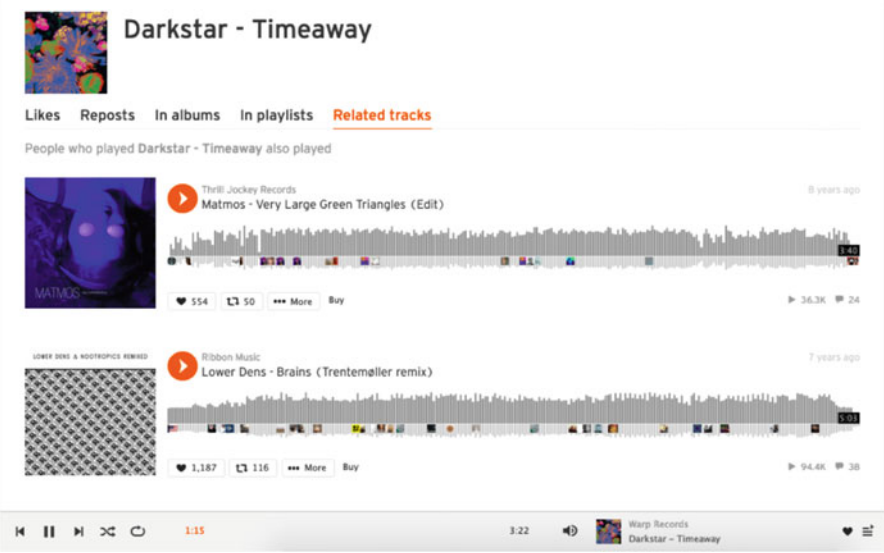

Fig. 1 Soundcloud's "Related tracks" web pages provide playlists of top 50 track recommendations for the tracks in their catalog. The list is pre-computed according to the seed in advance. The user is able to scroll and listen to the entire list, start and stop the playback (▶/❚❚), as well as navigate by clicking on a particular track or the next or previous track buttons (⏭/⏮)

iTunes[17] or Amazon[18]) and music metadata websites that allow users to organize their music collections (e.g., Discogs[19] or RateYourMusic[20]). They became much less common nowadays due to the advance of streaming platforms. Even in the shop scenario, explicit ratings may be too difficult to gather for the majority of the user base, and instead, systems rely on purchase history.

In the case of music streaming services, it is common to gather implicit user feedback. These systems often strive to minimize the required interaction effort while asking for explicit ratings can be tedious. Instead, they register each track played by a user (user listening events), compute play counts or total time listened for different items (tracks, albums, and artists), and track skips within the music player UI.

Such implicit feedback represents music preferences only indirectly. It can be dependent on user activity, context, and engagement, and there may be other reasons for user behavior unknown to the system. A played track in the user history does not necessarily mean the user actively liked it, and a skipped track does not necessarily

[17] https://www.apple.com/itunes.
[18] https://www.amazon.com.
[19] https://www.discogs.com.
[20] https://www.rateyourmusic.com.

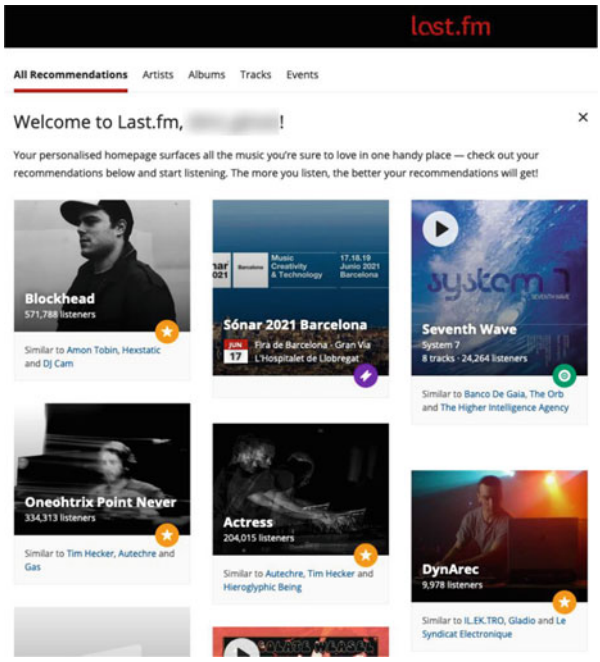

Fig. 2 Last.fm's landing page provides a list of personalized recommendations (artists, albums, tracks, and events) based on gathered listening statistics in a user profile. To give some context, the justification for the recommendations is provided by referencing similar music items listened by the user. The user can navigate the entire list as well as listen to a playlist with recommendations

imply negative preference. Still, this feedback is often the only information available to the system, and therefore it is used as a proxy for music preference.[21] Different criteria can be applied to consider a track as played and relevant for a user. For example, one can rely on the fraction of the total track duration that is reproduced within the system's UI and define a threshold to identify fully or almost fully played tracks. Also, the raw play count values can be normalized and thresholded to define relevant items (for example, in the simplest case, consider all items with at least one play as relevant).

2.1.2 Evaluation Metrics and Competitions

In essence, the basic recommendation task is the prediction of relevant items for a user and generation of recommendation lists with items ranked by relevance. The evaluation is commonly done in the offline setting, retaining part of the user

[21] Note that explicit ratings can be estimated from implicit feedback such as play counts, as investigated by Parra and Amatriain [137].

behavior data (e.g., user-item relevance ratings or play counts) as a ground truth and measuring the error in relevance predictions (typically via RMSE), or assessing the quality of generated ranked lists in terms of the position of relevant items therein, e.g. via precision at k, recall at k, mean average precision (MAP) at k, average percentile rank, or normalized discounted cumulative gain (NDCG). A few official research challenges described below have addressed these aspects.

Formulated as a purely collaborative filtering task, the problem has been addressed by the recommender systems community in the *KDD Cup 2011* challenge [44].[22] It featured a large-scale dataset of user-item ratings provided by *Yahoo!*[23] with different levels of granularity of the ratings (tracks, albums, artists, and genres) and a very high sparsity (99.96%) making the task particularly challenging. There were two objectives in the challenge, addressed on separate sub-tracks: predict unknown music ratings based on given explicit ratings (evaluated by RMSE) and distinguish highly rated songs from songs never rated by a user (evaluated by error rate). Unfortunately, the dataset for the challenge is anonymized, including all descriptive metadata, which made it impossible to try any approaches based on content analysis and music domain knowledge.

The *Million Song Dataset (MSD) Challenge*[24] [115] organized in 2012 opened the possibility to work with a wide variety of data sources (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 listening events for the test users. Mean average precision computed on the top 500 recommendations for each listener (MAP@500) was used as main performance measure.

2.2 Lean-in Exploration

Other music consumption settings emphasize more active and engaged user interaction. In these *lean-in* scenarios, the user is often exploring a collection and the found tracks in-depth to select candidates for listening immediately or at a later point. This can be used e.g. to find music that fits a certain affective state, activity, or setting such as a workout or a road trip. In many cases these scenarios are also tied to building and maintaining "personal" music collections within online platforms for individual or shared use, cf. [36, 37]. A recommender system can support such a process by presenting candidate tracks based on the user's behavior and adapting to the selection of tracks, cf. [80]. An example of a lean-in interface is Spotify's playlist creation interface, as shown in Fig. 3, in which recommendations are made to complement the tracks already added to a playlist.

[22] https://www.kdd.org/kdd2011/kddcup.shtml.

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

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

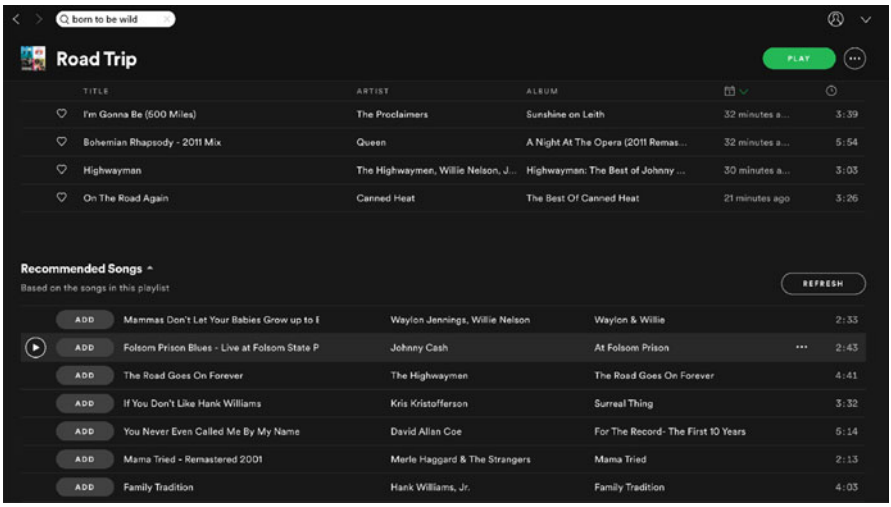

Fig. 3 Spotify's playlist creation application as an example of a *lean-in* experience. The user can search the catalog using the textual search box on top. Based on the songs added to the list, further songs are recommended at the bottom for consideration. Hovering over a track allows to play/pause the audio (▶/❚❚). Further recommendations can be requested by clicking the "Refresh" button. Before any songs are added, the user is prompted for a title and description for the playlist, which provide the basis for making initial recommendations. (Note that further playback control, navigation, advertisement, and social network panels in the interface are omitted in this screenshot.)

Lean-in-oriented interfaces provide a higher degree of control and are therefore richer in terms of user interface components, demanding more attention and a higher cognitive load from the user [126]. Controls often include search functionalities that add possibilities to retrieve specific tracks based on their metadata or via a "semantic" query that expresses the user intent, cf. Fig. 3. To index music pieces for semantic textual search, several sources can be tapped, such as knowledge graphs [134], various forms of community metadata such as tags or websites [90], or playlist titles given by other users [116]. This not only enriches the descriptions of individual tracks (e.g., to allow for queries like "90's band with female singer"), but also introduces information on context and usage purposes (e.g., "sleep" or "party").

Beyond playlist creation, lean-in interfaces have been proposed for a variety of tasks, e.g., exploration of musical structure, beat structure, melody line, and chords of a track [65], or exploration of tracks based on similar lyrics content [130]. These "active music-listening interfaces" [64], however, have not seen much adoption in commercial streaming platforms as they are often targeted at specific music consumers and non-traditional tasks, cf. [93].

2.2.1 Evaluation Metrics and Competitions

Evaluation metrics for lean-in scenarios are measuring similar aspects as in a less targeted browsing scenario, i.e. mostly retrieval-oriented metrics, see Sect. 2.1.2. In a focused task like the above mentioned playlist creation, however, feedback is available more explicitly, as selections are made from a known pool of tracks and a selection more strongly indicates a positive feedback as much as an omission a negative than is the case in general browsing. This also impacts evaluation, as information retrieval metrics like precision and recall can be meaningfully calculated.

This is reflected in the *2018 RecSys Challenge* [33],[25] which was centered on the task of recommending tracks for playlist creation, as performed in the Spotify application shown in Fig. 3. More specifically, provided with a playlist of specific length k (and optionally a playlist title indicating a context), the task was to recommend up to 500 tracks that fit the target characteristics of the given playlist. Different scenarios were addressed in terms of playlist types: by varying k, whether or not a playlist title was provided, and whether or not the playlist was shuffled. As part of the challenge, Spotify released a collection of 1 million user-generated playlists to be used for model development and evaluation.[26] Evaluation metrics used were R-precision, NDCG, and a Spotify-specific metric called "recommended songs clicks" (defined as the number of times a user has to request 10 more songs before the first relevant track is encountered). A detailed description and analysis of the top approaches can be found in [188].

2.2.2 Discussion

Lean-in experiences relate most closely to traditional directed information retrieval tasks like search and extend to all activities where a user is willing to devote time and attention to a system to enhance the personal experience. The role of the recommender system is to support the user in this specific scenario, e.g. by suggesting complementary items, without interfering, distracting, or persuading the user.

In terms of designing the recommendation algorithm, a lean-in scenario provides a good opportunity to favor *exploration* over *exploitation*. That is, the recommender system might not optimize for positive feedback only, but "probe" the user with potentially negatively perceived items. As such, these closer interactions between user and system provide an opportunity to develop the user profile for future recommendations, resulting in a longer-term reward than just the exploitation of items known to please the user (however, likely in a different context). Immediate

[25] https://www.recsyschallenge.com/2018.

[26] The *Million Playlist Dataset* is available from https://www.aicrowd.com/challenges/spotify-million-playlist-dataset-challenge.

user satisfaction does not necessarily suffer from this strategy, as more diverse recommendations are acceptable in a setting in which the user is open to reflect upon the suggestions made.

2.3 *Lean-Back Listening*

In contrast to the recommendation use cases outlined in the previous sections, the so-called *lean-back* formulation is designed to address use cases in which user interaction is minimized, such as automatic playlist generation or streaming radio. In lean-back settings, users typically are presented a single song at a time, which is selected automatically by the MRS. Often, it is expected that users do not have the interface directly in view, but rather are consuming recommendations on a smartphone application with the device out of view (e.g., in a pocket or bag, or while driving).

Typical lean-back user interfaces, such as the one depicted in Fig. 4 (left), tend to be minimal, and severely limit how the user can control the system. Although some

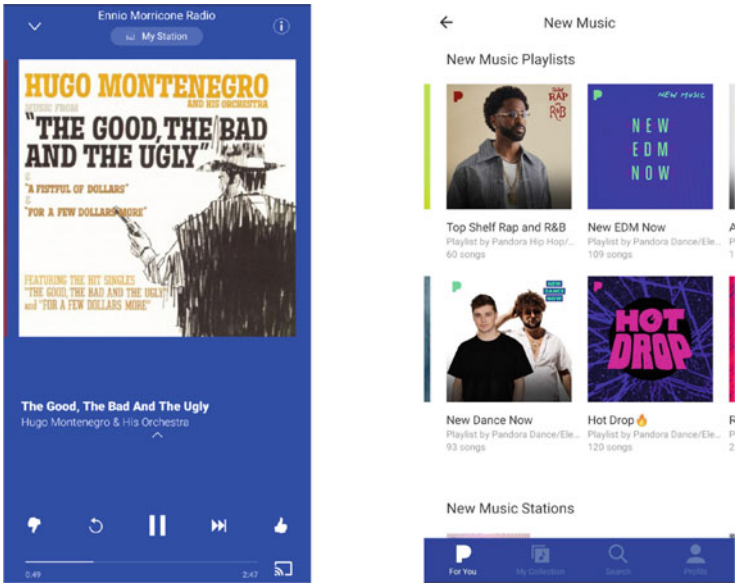

Fig. 4 Pandora's mobile application provides a prototypical example of a *lean-back* listening interface. Left: An image representing the current song (or an advertisement) is displayed, along with a progress bar and controls to switch "stations". The user can play/pause the audio (▶/❚❚), provide thumbs-up/down feedback (👍/👎), skip to the next track (⏭), or replay a track (↺). No information is provided for alternative track selections: the next track is selected automatically. Right: recommendations are organized by *stations* or *playlists*, allowing users to select a stream with minimal interactions

interfaces afford either positive or negative explicit feedback (likes, hearts, thumbs-up/thumbs-down), the fact that users are typically disengaged during consumption implies that explicit feedback is relatively rare. As a result, methods and evaluation criteria in lean-back settings tend to rely more upon implicit feedback, such as song completion, skipping, or session termination.

2.3.1 Lean-Back Data and Evaluation

Compared to the basic recommender system setup, the treatment of implicit feedback in lean-back settings requires a bit more nuance. In the basic setting, users are typically presented with a collection of items simultaneously, from which positive, negative, and relative interactions can be inferred efficiently. For example, if a user is recommended ten songs, and purchases only one, it is relatively safe to infer that the remaining nine were less relevant than the purchased song—if not outright *irrelevant*—which facilitates rapid and large-scale data collection [77]. Because lean-back interfaces do not present alternatives to the user, implicit feedback can only be inferred for a single item at a time, which significantly reduces the efficiency of data collection.

Lean-back music recommenders are often used in certain *contexts*: for example, while a user is exercising or working. Users are therefore expected to be inattentive, at least some portion of the time, and this can make it difficult to properly interpret even the weak signals that come from play, stop, and skip interactions. Accurately making sense of feedback gained in a lean-back setting is therefore highly challenging, and such data may not be suited to exhaustively model user preferences. For example, listening to a full song may be construed as a *positive* interaction, but it may also happen because a user was completely disengaged and forgot to turn off the stream. Alternatively, skipping a song may suggest a *negative* interaction, but a user may also skip a song that they otherwise like because they recently heard it elsewhere. Finally, a user may abandon a session because they are dissatisfied with the song selections (a negative interaction), or because they are finished with whatever outside activity they were performing. While these issues can be impossible to fully work around (barring invasive user surveillance), it is common to assume that play/skip/stop behavior, on average, provide weak positive and negative signals that can be used for model development and evaluation. However, these issues do highlight an important characteristic of lean-back recommendation: interactions take place within a particular *context*, and interpreting the resulting data outside its context can easily become problematic.

Lean-back interfaces are often designed around concepts of *playlists*, *radio stations*, or other similar abstractions which allow a user to express preference for groups of songs, e.g., by selecting a pre-generated playlist or station by description, or by choosing a genre, artist, or song to *seed* the session. This is depicted in Fig. 4 (right). As a result, much of the research on lean-back systems has focused on modeling playlists, which can be defined as either ordered or shuffled selections of songs that are meant to be heard together in a session. Playlists can be composed

by expert curators (like traditional disc jockeys), amateur users, or collaboratively by groups of users. Playlist authors often select songs deliberately to reflect some specific intended use or context [38]. Playlist data therefore provides an attractive source of high-quality positive interactions: co-occurrence of songs in a playlist can be a strong positive indicator of similarity. An algorithm which can accurately predict which songs should go into a playlist—perhaps conditioned on a context, user preference, or pre-selected songs—should therefore be useful for generating recommendations [59].

Note that modeling playlist *composition* is not equivalent to modeling playlist *consumption*. In the generation case, it is (perhaps arguably) justified to infer negative interactions: the playlist author presumably decided which songs to include. This in turn justifies the use of information retrieval metrics (precision, recall, etc.), which rely upon having a well-defined notion of what constitutes a *relevant* (included) or *irrelevant* (excluded) example. Playlist consumption, however, does not inherit relevance and irrelevance from playlist composition. The user was not exposed to alternative selections, so it is not justified to infer a negative interaction from tracks which were not included in the playlist. Evaluating a playlist generation method therefore relies on comparisons to playlist *authors*, not playlist *consumers*. In many commercial settings, authors can be employees of the service, while consumers are the customers, and it is important to bear this distinction in mind when developing a recommender system.

As an alternative to playlist data, *listening log* data can provide a more direct measurement of actual user behavior, as it captures how users behave in response to specific song selections. This makes listening log data attractive from a modeling perspective, but care must be taken to ensure that the data is interpreted correctly, and does not unduly propagate inductive bias from the system which selected the songs in the first place. That said, an algorithm which could accurately predict which songs are likely or unlikely to be skipped by a user (in a particular session or context) could be used to power a recommendation system. Developing and evaluating such a method requires large volumes of log data to capture the diversity of listening preferences and contexts. Fortunately, log data can be collected passively: unlike playlist generation, which requires deliberate intent from the playlist creator, log data is generated automatically by users interacting naturally with the system.

While user-generated playlist data has been relatively abundant and freely available [117, 139, 188], high-quality listening log data has been scarce outside of private, commercial environments. Until recently, the main source of openly available log data has been Last.fm scrobbles [28, 151, 181], which provide a large volume of data, but little in the way of transparency or provenance with respect to how tracks are selected and interacted with by users. This has recently changed with the MRS-related challenge in the *2019 WSDM Cup*:[27] participating teams had to predict whether a user will listen to or skip the music recommended next by a MRS. Participants were provided with a set of 130 million listening sessions by Spotify

[27] https://www.crowdai.org/challenges/spotify-sequential-skip-prediction-challenge.

users [23]. For each listening session, the first half of the session was observed, and the objective was to predict whether or not each track in the second half of the session was actually consumed or skipped. The adopted evaluation metric was mean average accuracy, where average accuracy was computed over all items in the unseen part of each session. The winning approaches are described in [32, 69, 195].

2.3.2 Discussion

In contrast to the basic recommendation formulation, lean-back settings require careful attention to several unique factors arising from the characteristics of music consumption.

First, as previously mentioned, users in lean-back scenarios are assumed to want minimal interaction with the recommender system. Concretely, lean-back music recommenders are often designed to be more conservative with recommendations, prioritizing *exploitation* over *exploration* to minimize negative feedback (i.e., skips) [29]. While the exploration–exploitation trade-off is a well-known concept in recommender systems generally [163], one must bear in mind that the right trade-off fundamentally depends on the mode of delivery and interaction with the user.

Second, the sequential nature of lean-back recommendation scenarios presents a substantial methodological challenge to evaluation. For example, skip prediction methods are trained and evaluated on historical log data, which is almost always biased by whichever recommendation algorithm was used at the time of the interactions. Left unchecked, this can propagate inductive bias from previous algorithms, and skew the evaluation results. While not inherently unique to the lean-back setting, the rapid sequential consumption of recommendations in this context renders typical simplifying assumptions (e.g., independence between interactions) suspicious at best. Compensating for this source of bias is generally challenging, though in some situations, counterfactual risk minimization [170] can be employed during training [120].

Finally, playlist data may carry biases beyond what are commonly found in standard collaborative filter data. In particular, several streaming platforms employ content curators to create playlists which can be shared to users, or allow (and promote) users to share playlists with each other. While some amount of curation is undoubtedly desirable in many situations, it is also important to understand the sources of bias in artist and track selection within the data when developing and evaluating a playlist generation algorithm.

2.4 Other Applications

There also exist various applications beyond the main lines of research on music recommendation. In particular, *music event recommendation* is addressed in [176], where the authors consider recommending events/venues with local long-tail artists,

from which metadata available to the system may be lacking. A related task is the recommendation of music for particular venues, addressed in [35]. Last.fm and Spotify are examples of industrial systems that provide event recommendations.

Playlist discovery and *playlist recommendation* are other recent directions [136]. They have not yet received as much attention as playlist continuation (Sect. 2.3): even though there is research on track recommendation for playlists and listening sessions, a lack of studies on recommending entire playlists to a user is evident.

We can also highlight the future role of recommender systems in music and video production. An exemplary use case here is *recommending background music for video*. The goal of this task is to assist the user in finding music that fits video scenes in their video production in terms of semantics, rhythm, and motion. Such systems rely on multi-modal analysis of audio and video [100, 110]. In turn, in music production, recommender systems can enrich the users' workflow, helping navigate extensive audio collections. In particular, *sound recommendation* has been considered in [134, 165]. Based on audio analysis and domain knowledge in music composition, recommender systems will open promising possibilities for building more intelligent digital audio workstations with sound, loop, and audio effect recommendation functionalities.

Another task related to MRS is to *recommend digital score sheets*, such as implemented in the system offered by OKTAV[28] for piano players; or to *recommend chords* for guitar players, offered by Chordify.[29]

3 Types of Music Recommender Systems

In the following, we briefly characterize the major types of MRS and summarize the input data and techniques adopted in each type. Please note that the research works we point to by no means represent an exhaustive list. We intentionally kept this general part rather short and point the reader, for instance, to our chapter in the second edition of the Recommender Systems Handbook [159] for a more detailed description of the different types of MRS.

3.1 Collaborative Filtering

Similar to other recommendation domains, CF-based approaches are often used for music recommendation. They operate solely on data about user–item interactions, which are either given as explicit ratings (e.g., on a rating scale) or as implicit feedback (e.g., statistics on play counts or skipped songs). In the most common

[28] https://www.oktav.com.

[29] https://www.chordify.net.

variant, CF approaches create a model that predicts whether a given user will interact or not with a (previously unseen) item. Since approaches solely based on CF are domain-agnostic, we invite the reader to consider the surveys provided in [34, 73, 86, 96] for a more detailed general treatment of the topic; and the surveys provided in [152, 161] that review CF approaches in the music domain, among others. Also consider chapter "Advances in Collaborative Filtering" of this book.

Note that CF-based approaches are particularly prone to several kinds of biases, such as *data bias* (e.g., community bias and popularity bias) and *algorithmic bias*. *community bias* refers to a distortion of the data (user–item interactions) caused by the fact that the users of a certain MRS platform do not form a representative sample of the population at large. An example is provided by a study of classical music on Last.fm and Twitter, carried out in [160]. The over- or under-representation of a user group, and in turn resulting user preferences, influences the quality of CF-based algorithms. *Popularity bias* occurs when certain items receive many more user interactions than others. This relates to the long-tail property of consumption behavior and can favor such highly popular items [5, 98]. Other data biases and artifacts in music usage data stem from additional factors such as record label associations [89].

In terms of *algorithmic bias*, Ekstrand et al. [48] find an effect of age and gender on recommendation performance of CF algorithms, even when equalizing the amount of data considered in each user group. Similarly, Melchiorre et al. [124] identify considerable personality bias in MRS algorithms, i.e., users with different personalities receive recommendations of different quality levels which depend on the adopted recommendation algorithm. To alleviate these undesirable effects, devising methods for debiasing is one of the current big challenges in MRS research (see Sect. 4.3).

3.2 Content-Based Filtering

While CF-based approaches operate solely on user–item interaction data, the main ingredient to CBF algorithms is content information about items, which is used to create the user profiles. To compute recommendations for a target user, no other users' interaction data is needed.

Recommendations are commonly made based on the similarity between the user profile and the item representations in a top-k fashion, where the former is created from individual statistics of item content interacted with by the user. In other words, given the content-based user profile of the target user, the items with best matching content representations are recommended. Alternatively, a machine learning model can be trained to directly predict the preference of a user for an item.

A crucial task for every CBF approach is the representation of content information, which is commonly provided in form of a feature vector that can be composed of (1) handcrafted features or (2) latent embeddings from deep learning tasks such as auto-tagging. In the former case, commonly leveraged features include

computational audio descriptors of rhythm, melody, harmony, or timbre, but also metadata such as genre, style, or epoch (either extracted from user-generated tags or provided as editorial information), e.g. [20, 21, 42, 66, 118]. We refer to [128] for an overview of music audio descriptors typically used in MIR. As for the latter (2), latent item embeddings are often computed from low-level audio signal representations such as spectrograms, e.g. [178], or from textual metadata such as words in artist biographies, e.g. [133]. Higher-level semantic descriptors (such as genres, moods, and instrumentation) retrieved from audio features by machine learning can also be used and have been shown to correlate with preference models in music psychology [60].

One of the earliest and most cited deep-learning-based approached to CBF in the music domain is van den Oord et al.'s work [178]. The authors train a convolutional neural network (CNN) to predict user–item latent factors from matrix factorization of collaborative filtering data. The resulting CNN is then able to infer these latent representations from audio only.

More recent deep learning approaches based on content representations include [74, 109, 133, 148, 177, 177]. To give one example, Vall et al. [177] propose a content-based approach that predicts whether a track fits a given user's listening profile (or playlist). To this end, all tracks are first represented as a feature vector (constituting of, e.g., text embeddings of Last.fm tags or audio features such as i-vectors based on MFCCs [47]). These track vectors are subsequently fed into a CNN to transform both tracks and profiles into a latent factor space; profiles by averaging the network's output of their constituting tracks. As common in other approaches too, tracks and user profiles are eventually represented in a single vector space, which allows to compute standard distance metrics in order to identify the best fitting tracks given a user profile.

3.3 Hybrid Approaches

There exist several perspectives as to what makes a recommendation approach a hybrid one: either (1) the consideration of several, complementary data sources or (2) the combination of two or more recommendation techniques.[30] In the former case, complementary data sources can also be leveraged when only a single recommendation technique is used. For instance, a CBF-based MRS can exploit both textual information (e.g., tags) and acoustic clues (e.g., MFCC features), cf. [41].

As for the latter perspective, i.e., combining two or more recommendation techniques, a common strategy is to integrate a CBF with a CF component. Traditionally, this has often been achieved in a late fusion manner, i.e., the recommendations

[30] Note that perspective (2) most commonly also entails (1) since different recommendation techniques require different data to operate on.

made by two separate recommendation models are merged by an aggregation function to create the final recommendation list. Examples in the area of music recommendation include [84, 112, 114, 171]. Tiemann and Pauws [171] propose an item-based memory-based CF and an audio-based CBF that make independent rating predictions, which are aggregated based on rating vector similarities. Lu and Tseng [112] fuse the output of three nearest-neighbor (top-k) recommenders: two CBF approaches based on similarity of musical scores and of emotion tags, and one CF approach; the final recommendation list is then created by reranking items based on personalized weighting of the three components. Mcfee et al. [114] optimize a content-based similarity metric (based on MFCC audio features) by learning from a sample of collaborative data. Kaminskas et al. [84] use Borda rank aggregation to combine into a single recommendation list the items recommended by an auto-tagging-based CBF recommender that performs matching via emotion labels and a knowledge-based approach that exploits DBpedia.[31] For a comprehensive treatment of hybridization techniques, we refer to [25, 75].

In contrast, most current deep learning approaches integrate into a deep neural network architecture audio content information and collaborative information such as co-listening or co-rating of songs or artists, e.g. [74, 133]. Furthermore, these approaches can incorporate other types of (textual) metadata. For instance, Oramas et al. [133] first use weighted matrix factorization to obtain, from users' implicit feedback (track play counts), artist and track latent factors. These latent factors are used as a prediction target to train two neural networks: one to create track embeddings, the other to obtain artist embeddings, exploiting spectrograms and biographies, respectively. Based on the spectrograms (constant-Q transformed), a CNN is used to create track embeddings. Biographies are represented as TF-IDF vectors and a multilayer perceptron (MLP) is trained to obtain the latent artist embeddings. Eventually, the resulting track and artist embeddings are concatenated and fed into another MLP, trained to predict final track latent factors. To create ranked recommendations, the dot product between a given user latent factor and the final track factors is computed.

3.4 Context-Aware Approaches

Definitions of what constitutes the "context" of an item, a user, or an interaction between the two are manifold. So are recommendation approaches that are named "context-based" or "context-aware". For a meta-study on different taxonomies and categories of context, see for instance [16]. Here, in the context of MRS, we adopt a pragmatic perspective and distinguish between item-related context, user-related

[31] https://wiki.dbpedia.org.

context, and interaction-related context.[32] Item-related context may constitute, for instance, of the position of a track in a playlist or listening session. User-related context includes demographics, cultural background, activity, or mood of the music listener. Interaction-related or situational context refers to the characteristics of the very listening event, and include aspects of time and location, among others.

There exist various strategies to integrate context information into a MRS, which vary dependent on the type of context considered. Simple variants include *contextual prefiltering* and *contextual postfiltering*, cf. [7]. In the former case, only the portion of the data that fits the user context is chosen to create a recommender model and effect recommendations, e.g. [17, 157]; or users or items are duplicated and considered in different recommendation models if their ratings differ for different contexts, e.g. [15, 194]. In contrast, when adopting contextual postfiltering, a recommendation model that disregards context information is first created; subsequently, the predictions made by this model are adjusted, conditioned on the context, to make the final recommendations, e.g. [193].

An alternative to contextual filtering approaches is to extend latent factor models by contextual dimensions. If user–item interactions are provided "in context" and differ between contexts, a common approach is to extend matrix factorization to *tensor factorization*, i.e., instead of a matrix of user–item ratings, a tensor of user–item–context ratings is factorized, so that each item, user, and context can be represented. More details and examples can be found, among others, in [1, 14, 62, 85].

Recently, deep neural network approaches to context-aware MRS have emerged. They often simply concatenate the content- or interaction-based input vector to the network with a contextual feature vector, e.g. [182]. Another approach is to integrate context through a gating mechanism, e.g., by computing the element-wise product between context embeddings and the neural network's hidden states [19]. An example in the music domain is [153], where the authors propose a variational autoencoder architecture extended by a gating mechanism that is fed with different models created from users' country information.

3.5 Sequential Recommendation

Sequence-aware recommender systems play a crucial role in music recommendation, in particular for tasks such as *next-track recommendation* or *automatic playlist continuation* that aims at creating a coherent sequence of music items. Corresponding approaches consider sequential patterns of songs, e.g., based on playlists or listening sessions, and create a model thereof. Note that such approaches can also be considered a variant of context-aware recommendation in which item

[32] Note that we use the term "interaction data" in Sect. 4 to refer to data belonging to the latter kind of context.

context is leveraged in form of preceding and subsequent tracks in the sequence. To create such a system, most state-of-the-art algorithms employ variants of recurrent or convolutional neural networks, or autoencoders [152, 188]. For a more detailed treatment of the subject matter, we refer the reader to recent survey articles on sequence-aware recommendation, which also review approaches to sequential music recommendation, e.g. [143, 183].

3.6 Psychology-Inspired Approaches

Recently, research on recommender systems emerged that aims at enhancing the traditional data-driven techniques (based on user–item interactions like in CF or item content information like in CBF) with psychological constructs. Examples include the use of models of human memory, personality traits, or affective states (mood or emotion) of the user in the recommendation algorithm. There exist several psychological models to formalize human memory, including the adaptive control of thought-rational (ACT-R) [10] and the inverted-U model [131], which have been studied in the context of music preferences. These also relate to familiarity and novelty aspects discussed in Sect. 4.1.2.

For instance, Kowald et al. [97] propose an approach to MRS that integrates a *psychological model of human memory*, i.e., ACT-R [10]. They identify two factors that are important for remembering music: (1) frequency of exposure and (2) recentness of exposure. Their ACT-R-based approach outperforms a popularity baseline, several CF variants, and models that only consider one of the two factors mentioned above.

MRS approaches that consider the users' *personality traits* rely on insights gained from studies that relate personality traits to music preferences, e.g. [45, 56, 123, 144, 145, 155]. Building upon such work, Lu and Tintarev in [113] propose a MRS that adapts the level of diversity in the recommendation list according to the personality traits of the user, by reranking the results of a CF system. The proposed MRS builds upon their finding that users with different personalities prefer different levels of diversity in terms of music key, genres, and artists. Another personality-aware approach to music recommendation (and recommendation in other media domains) is presented by Fernández-Tobías et al. in [50]. The authors integrate into a traditional matrix factorization-based CF approach a user latent factor that describes the user's personality traits.

Research on MRS that considers the user's *affective state*, such as mood or emotion, when computing a recommendation list rely on results of studies in music psychology and cognition that identified correlations between perceived or induced emotion on the one hand, and musical properties of the music listened to on the other hand, e.g. [46, 72, 79, 174, 190]. Such insights are exploited in MRS, for instance in [12, 43]. Deng et al. [43] acquire the users' emotional state and music listening information by applying natural language processing techniques on a corpus of microblogs. Leveraging the temporal vicinity of extracted emotions and

listening events, the authors consider emotion as a contextual factor of the user–song interaction. They integrate this contextual information into a user-based CF model, an item-based CF model, a hybrid of the two, and a random walk approach. Ayata et al. [12] propose a MRS architecture that uses the affective response of a user to the previously recommended songs in order to adapt future recommendations. Their system leverages data from wearable sensors as physiological signals, which are used to infer the user's emotional state.

For a comprehensive survey on psychology-informed recommender systems, we refer the reader to [108].

4 Challenges

In the following, we provide a discussion of major challenges that are faced in MRS research and practice, and present approaches to address these challenges.

4.1 How to Ensure and Measure Multi-Faceted Qualities of Recommendation Lists?

The music items that constitute the recommendation list of a MRS should fulfill a variety of quality criteria. Obviously, they should match the user's preferences or needs. Additional criteria, depending on the situational context or state of the listener (cf. lean-in and lean-back tasks in Sect. 2), are equally important, though. In the following, we identify and discuss several of these characteristics that contribute to a good recommendation list.

4.1.1 Similarity Versus Diversity

Items in the recommendation list should be similar to the user's preferred tracks, and also show a certain extent of similarity between them. Determining similarity of music items is a multi-faceted and non-trivial, if not actually elusive, task [90, 91]. However, operational models of acoustic similarity have been used in the past.[33] Notwithstanding all optimization for similarity and consistency, it has been shown that music recommendation lists containing only highly similar items are often perceived as boring [104]. At the same time, diversity in recommendation lists has a positive impact on conversion and retention [9]. Therefore, the right balance between similarity and diversity of items is key to optimize user satisfaction. Note that the level of diversity (and accordingly similarity) itself can be personalized,

[33] https://www.music-ir.org/mirex/wiki.

e.g., using a linear weighting on similarity and diversity metrics. In fact, it has been shown that different users prefer different levels of diversity [172, 186].

An overview of diversification strategies in recommender systems is given in [27]. A common approach to measure diversity is to compute the inverse of average pairwise similarities between all items in the recommendation list [166, 196]. In particular in the music domain, similarity and diversity are highly multi-faceted constructs; user preferences for them should therefore also be investigated and modeled in a multi-faceted manner. To give an example, a user may want to receive recommendations of songs with the same rhythm (similarity) but from different artists or genres (diversity).

4.1.2 Novelty Versus Familiarity

On the one hand, a recommendation list should contain content the user is familiar with, e.g., known songs or songs by a known artist, not least because familiarity seems vital to engage listeners emotionally with music [138]. On the other hand, a recommendation list should typically also contain a certain amount of items that are novel to the target user. This can be easily measured as the fraction of artists, albums, or tracks in the user's recommendation list that the user has not been interacted with or is not aware of [27, 156]. However, whether or not a user already knows a track is not always easy to determine. A recommended item—novel according to the data—may, in contrast, be already known to the user. For instance, a user may have listened to the item on another platform. In this case, the MRS does not know that the item is, in fact, not novel to the user. On the other hand, a user might have forgotten a track that he or she has not listened to for a while, i.e., the user is (no longer) familiar with the content.[34] In this case, the track may be a novel and interesting recommendation for the user, even though the user has already interacted with it.

An interesting insight into users' preferences for novelty has been gained in [184], where Ward et al. found that even for users who indicate that they prefer novel music, familiarity is a positive preference predictor for songs, playlists, and radio stations.

4.1.3 Popularity, Hotness, and Trendiness

In contrast to novelty and familiarity, which are defined for individual users (is a particular user already familiar with an item?), popularity, hotness, or trendiness refer to aspects that are global measures, commonly computed on system level. These terms are often used interchangeably; though, sometimes hotness and trendiness refer to a more recent or current scope, related to music charts, while popularity is considered

[34] This scenario is addressed in MRS that leverage cognitive models of frequency and recentness of exposure, discussed in Sect. 3.6.

time-independent. Offering track lists created by a popularity-based recommender is a common feature of many commercial MRS, to mitigate cold start or keep the user up-to-date about trending music. Even though such lists are not personalized, they allow to engage users, serving as an entry point to the system and as basic discovery tool. For this reason, studies in MRS often consider recommendations based on popularity as a baseline.

A concept related to popularity is "mainstreaminess" [17], also known as "mainstreamness" [180], which is sometimes leveraged in MRS. Assuming that users prefer to different extents music that is considered mainstream, such systems tailor the amount of highly popular and of long-tail items in the recommendation list to the user's preference for mainstream music. What constitutes the music mainstream can further be contextualized, for instance, by defining the mainstreaminess of an item or user at the scope of a country or an age group [17, 180].

4.1.4 Serendipity

The topic of serendipity has attracted some attention a few years ago, but less nowadays. Serendipity is often defined rather vaguely as items being relevant and useful but at the same time surprising to or unexpected by the user [161]. The notion of unexpectedness, which is central to serendipity, is often interpreted as being unknown to the user or far away from the user's regular music taste [82]. For instance, in their proposal for a serendipitous MRS, Zhang et al. use the average similarity between the user's known music and the candidates for song recommendation to define a measure of "unserendipity" [191].

4.1.5 Sequential Coherence

The coherence of music items in the recommendation list is another qualitative aspect of a MRS. What is considered a coherent sequence of songs in a listening session or playlist is highly subjective and influenced by individual preferences, though [104]. Some findings on aspects of coherence that recur in different studies include a common theme, story, or intention (e.g., soundtracks of a movie, music for doing sports), a common era (e.g., songs of the 1990s), and the same (or similar) artist, genre, style, or orchestration [38, 40, 104].

Focusing on music playlists, recent studies conducted by Kamehkhosh et al. investigated the criteria that are applied when music lovers create playlists, either with or without support by a recommender engine [80, 81]. In a study involving 270 participants, the following characteristics of playlists were judged by the participants according to their importance: homogeneity of musical features such as tempo, energy, or loudness (indicated as important by 64% of participants), diversity of artists (57%), lyrics' fit to the playlist's topic (37%), track order (25%), the transition between tracks (24%), tracks popularity (21%), and freshness (20%).

4.2 How to Consider Intrinsic User Characteristics?

Modeling the user is central to providing personalized recommendations. Traditionally, a user model for recommender systems consists of the history of recorded interactions or expressions of preference (cf. Sect. 2.1), a latent space embedding derived from these, or meta-features aggregating usage patterns based on domain knowledge, such as "exploratoryness", "genderedness", "fringeness", or the above mentioned "mainstreamness" [17, 181]. Describing and grouping users based on such dimensions then allows to tailor recommendations accordingly, e.g. by balancing similarity and diversity, cf. Sect. 4.1.1. However, in a cold-start setting, this information can not be resorted to. In addition, one might argue that none of these descriptions model "the user" in a task- or domain-independent manner, i.e. incorporate intrinsic user characteristics.

In order to avoid user cold-start and include individual user information into models, external data can be used. Such individual aspects cover *demographic information*, such as age or sex, which have been shown to have an impact on music preference. For instance, there is evidence that younger users explore a larger number of genres, while, with increasing age, music preferences become more stable [55, 144]. A further direction in this line of research is to make use of psychological models of *personality* to inform the recommender system, also see chapter "Personality and Recommender Systems". Personality is a stable, general model relating to the behavior, preferences, and needs of people [78] and is commonly expressed via the five factor model, based on the dimensions openness to experience, conscientiousness, extraversion, agreeableness, and neuroticism. For interaction with online music systems, personality has been related to music browsing preferences [57] and diversity needs [53], among others. Another stable attribute to describe users is to identify them from a *cultural* point of view, i.e. by associating them with their country or culture, and building upon aggregated usage patterns [17].

More dynamic aspects of user characteristics concern *affect and emotional state* and *listening intent* of the user. Short-time music preference, i.e. what people want to listen to in a specific moment, depends strongly on the affective state of the user, e.g. [83]. This, again, can partly be traced to questions of personality, e.g. there is indication that when being sad, open and agreeable persons prefer happy music, whereas introverts prefer sad music [54]. Vice versa, music influences the user's affective state. Affect regulation is therefore not surprisingly one of the main listening intents people have [95, 111, 132, 149]. Other important motivations for listening to music are, obviously, discovery and serendipitous encounters [39, 101] and social interaction [87]. Differences in goals and intended outcomes do not only impact the type of music recommender to be built, but also the type of evaluation strategy and success criteria to be chosen (see Sects. 2 and 4.5). To facilitate this line of research further, a deeper understanding of how people seek and discover new music in their everyday life, mediated through online platforms is needed. Pointers can be found in existing work utilizing user-centric evaluations, in-situ

interviews, and ethnographic studies, cf. [39, 102, 103, 105, 106, 168]. A more thorough discussion on user aspects in music recommendation can be found in [92].

To include models of user characteristics into a recommendation systems, the more static dimensions providing demographic and personality information can be treated as side information or even as fixed-target user dimensions in collaborative filtering matrix factorization approaches. Other, feature-driven methods like factorization machines or deep neural networks can benefit by incorporating this information as user dimensions. Section 3.6 highlights existing work that integrates these dimensions into MRS. The more dynamic dimensions of user affect and intent can also be incorporated as contextual information, cf. Sect. 3.4.

4.3 How to Make a Music Recommender Fair?

The increasing adoption of machine learning methods and data-driven recommender systems in real-world applications has exhibited outcomes discriminating against protected groups and enforced stereotypes and biases (cf. Sect. 3.1). These outcomes are considered unfair, as they treat individuals and groups of people differently based on sensitive characteristic (or even violate anti-discrimination laws), and irrespective of personal preference, cf. [26]. To address this, existing work builds upon operational definitions of fairness emphasizing parity at different machine learning stages [179] and aims at improving fairness by overcoming inadequacy of metrics and bias in datasets, optimization objectives, or evaluation procedures [187].

To address the question of fairness in MRS, the bigger picture of the music industry ecosystem needs to be taken into account, cf. [4, 158]. As are other recommendation domains, music recommendation is a multi-stakeholder setting, in which different stakeholders have different, contradictory goals [3, 24]. Stakeholders in the music domain include the listeners, the artists, the composers, the right owners, the publishers, the record labels, the distributors, and the music streaming services, to mention but the most important. The question of fairness therefore first needs to be phrased as "fairness for whom"?

Although a large number of stakeholders is involved in the recommendation process, fairness in music recommendation is often reduced to being a two-sided setting where fairness towards artists (as providers of the items) and the users (as consumers) needs to be maintained [122]. Even in this reduced setting, the opportunity of the artists as item providers, i.e. fairness with respect to exposure to the customer, is prioritized over fairness for users. This can be attributed to a fairness definition for users which is simply equated with high user satisfaction and prediction accuracy, e.g. by calibrating for diversity in user histories [122, 169]. A common restraint for such a definition of user fairness can be found in item popularity [6, 98]. This two-sided view, however, neglects the very important role of the recommender system platform as an intermediary between provider and consumer, cf. [2]. Non-neutrality of the platform, due to utility associated with

recommending certain items, presents another potential source of unfairness to both artists and listeners, cf. [88].

It is obvious that in order to optimize MRS towards fairness, several perspectives and notions can be adopted and need to be taken into consideration. Investigating perceptions and definitions of fairness for different tasks and different stakeholders is therefore a highly relevant question for future research.

4.4 How to Explain Recommendations?

Providing justifications of recommendations offer the user a means to understand why the system recommended a certain item. It has been shown that this can increase user trust and engagement [11, 61, 146, 173]. Depending on the recommendation task and technique, such justifications can include similar users' preferences, predicted rating values, common properties between liked items and recommendations (e.g., "We are recommending this song because you seem to like Viking Metal.") [125, 129], or natural language explanations based on item and/or contextual features (e.g., "We are recommending this song because it is energetic and will stimulate your current sports activity.") [31].

Early works on the topic of explainability in the music domain commonly adopt a content-based approach to explain artist or track similarities. For instance, Green et al. [67] use social tags and Wikipedia descriptions to create a tag cloud explaining artist similarities. Likewise, Pampalk and Goto [135] integrate user-generated artist tags into a music recommendation and discovery system, thereby enabling users to steer the artist recommendation process. Turnbull et al. use an auto-tagger to predict semantic labels from audio signal features in order to generate verbal descriptions of tracks [175].

More recent examples of explainability in MRS include *Moodplay* [11], a recommendation and exploration interface which builds upon a visualized latent mood space created from artist mood tags. Andjelkovic et al. integrate a CBF recommender based on acoustic artist similarity into the audiovisual interface. This interface can be used both to interact with and to explain the recommended artists. Another method is BAndits for Recsplanations as Treatments (*Bart*) by Spotify [120]. Adopting a reinforcement learning strategy, *Bart* learns interactions between items and explanations conditioned on the user or item context. User engagement as result of an explanation is used as a reward function. Explanations include, for instance, time ("Because it's Monday"), novelty ("Because it's a new release"), genre ("Because you like Jazz"), or popularity ("Because it's popular"). McInerney et al. also find that personalizing not only the recommendations, but also the explanations substantially increase user satisfaction.

While research on explainability in MRS is still scarce, though recently seeing a strong increase, a vital aspect to keep in mind is that users differ in their demand for and acceptance of explanations. In fact, Millecamp et al. [125] find that personal characteristics such as musical expertise, tech savviness, or need for cognition

influence how users interact with MRS and perceive explanations. Therefore, a vital challenge to address is not only to devise methods for explaining recommendation results, but also tailor these methods to the needs of the MRS' users.

4.5 How to Evaluate a Music Recommender System?

In the recommender systems literature, evaluation strategies are commonly divided into *offline testing*, *online testing*, and *user studies*, cf. [68]. In the following, these are briefly discussed in the context of MRS. On a critical note, most if not all of the methods discussed below do not consider bias and fairness, cf. Sect. 4.3.

4.5.1 Offline Evaluation

Offline evaluation relies on existing datasets of user–item interactions, and is carried out without involvement of users during evaluation. In this way, it enables gaining quantitative and objective insights into the performance of preference prediction algorithms, and is similar to quantitative evaluation strategies found in the machine learning and (music) information retrieval literature [13, 161].

Like in offline evaluation of other types of recommender systems, *performance metrics* commonly used when evaluating MRS include *error measures* such as root-mean-squared error (RMSE) computed between predicted and true ratings,[35] *item relevance measures* such as recall and precision, and *rank-based metrics* such as mean average precision (MAP), normalized discounted cumulative gain (NDCG), and mean percentile rank (MPR). More details can be found in [161], for instance. In addition to these measures of accuracy, beyond-accuracy measures that gauge some of the characteristics described in Sect. 4.1 are tailored to the music domain and include metrics for diversity, familiarity, popularity, and serendipity.

Public *datasets* available for academic research are summarized in Tables 1 and 2. Table 1 contains basic statistics of the datasets, such as number of songs, albums, artists, users, and user–item interactions. Table 2 provides more details on the composition of each dataset, including release year, origin of the data, and the kinds of item-, user-, and interaction-related data that is included: interaction data (e.g., ratings or timestamps), item data (e.g., tags), and user data (e.g., demographics). Note that the available datasets are dominated by industrial data released for research. There is no publicly gathered data except for the ListenBrainz initiative that strives for building an open alternative to gathering listener behavior

[35] Note that these ratings can also be binary (1 if the user interacted with the item; 0 otherwise).

Table 1 Statistics of public data sets for music recommendation research

Dataset	Songs	Albums	Artists	Users	Interactions
Yahoo! Music [44]	–	625K in total	–	1M	262.81M
MSD [18]	1M			1.02M	48.37M
Last.fm–360K [28]			187K	359K	
Last.fm–1K [28]			108K	1K	19.15M
MusicMicro [150]	71K		20K	137K	594K
MMTD [70]	134K		25K	215K	1.09M
AotM-2011 [117]	98K		17K	16K	859K
LFM-1b [151, 154]	32M	16M	3M	120K	1B
MSSD [23]	3.7M				150M
MLHD [181]	7M	900K	555K	583K	27B
ListenBrainz[a]	>7.58M	>1.32M	>776K	15K	507.84M
MPD [188]	2.26M	735K	296K		
Spotify Playlists [139]	1.88M		277K	15K	144K
#nowplaying [189]	1.21M			4.15M	46.05M
#nowplaying-RS [140]	346K			139K	11.64M
Melon Playlist Dataset [52]	649K	269K	108K		

[a] https://listenbrainz.org, statistics as of January 3, 2022. Only tracks mapped to MBIDs with direct string matching are reported

data. As a result, all these dataset have a clear bias towards Western music as they primarily originate from Western companies.[36]

Following offline strategies is still the predominant way of evaluating MRS in academia, not least due to the lack of contacts to (large numbers of) real users. However, despite their obvious advantages, offline evaluations do not provide sufficient clues on the perceived quality of recommendations and their actual usefulness for the listener [162], and there is research evidence that high recommender accuracy does not always correlate with user satisfaction [121]. They also do not account for biases on neither the consumer nor the artist side [98]. Furthermore, if recommender systems are targeted towards discovery, it is fundamental to assess the listener's familiarity with the recommended items apart from their relevance, which is problematic using existing datasets. Another critical point of offline studies is the overly high confidence in results, often seen in publications. This probably arises due to the computational and seemingly "objective" nature of the metrics. With huge amounts of evaluation data on user–item interactions available, it has become common to train the recommendation models and compute evaluation metrics only on a randomly drawn sample of the data, in particular to select negative/irrelevant items. However, results obtained by this kind of evaluation often show low variance (when metrics are computed across different sets of random samples), but high

[36] The Melon Playlist Dataset is a notable exception, containing data from a South Korean music streaming service.

Table 2 Features of public data sets for music recommendation research. The following symbols are used to denote the featured categories of data: ♪ single listening events, ♫ playlists or listening sessions, 👍 ratings, 🏷 tags, ♫🏷 playlist titles or annotations, 🔊 audio features, 🕐 temporal information/timestamps, 📍 location, ⚥ gender, 👤+ age, MusicBrainz or Twitter identifiers. Please note that we always denote the entity for which the data is provided, e.g., user-generated tags can be provided as *item data* (ignoring the information about which users provides them), but also as *interaction data* (indicating which user assigned which tag to which item)

Dataset	Year	Source	Interaction data	Item data	User data
Yahoo! Music [44]	2011	*Yahoo*	♪ 👍	✗	✗
MSD [18]	2011	*Echo Nest*	♪	MB 🏷 🔊	✗
Last.fm – 360K [28]	2010	*Last.fm*	♪	MB	📍 ⚥ 👤+
Last.fm – 1K [28]	2010	*Last.fm*	♪	MB	📍 ⚥ 👤+
MusicMicro [150]	2013	*Twitter*	♪ 🕐 📍	MB	MB
MMTD [70]	2013	*Twitter*	♪ 🕐	MB	MB
AotM-2011 [117]	2011	*Art of the Mix*	♫🏷 🕐	MB 🏷 🔊	🕐
LFM-1b [151, 154]	2016	*Last.fm*	♪ 🕐	🏷	📍 ⚥ 👤+
MSSD [23]	2019	*Spotify*	♫ 🕐	MB 🔊	✗
MLHD [181]	2017	*Last.fm*	♪	MB	📍 ⚥ 👤+
ListenBrainz	2015-	*ListenBrainz*	♪	MB	✗
MPD [188]	2018	*Spotify*	♫🏷	MB	✗
Spotify Playlists [139]	2015	*Spotify*	♫🏷	✗	✗
#nowplaying [189]	2014	*Twitter*	♪ 🕐	MB	✗
#nowplaying-RS [140]	2018	*Twitter*	♪ 🕐 🏷	🔊	MB
Melon Playlist Dataset [52]	2021	*Melon*	♫🏷	🔊 🏷	✗

bias, the former being particularly dangerous as it is likely to lead to an unjustified confidence in evaluation results [99].

4.5.2 Online Evaluation

While offline evaluation is still the predominantly adopted evaluation methodology in academia, evaluation of MRS in industrial settings is nowadays dominated by online studies, involving real users. This, of course, does not come as a surprise since MRS providers such as Spotify, Deezer, or Amazon Music have millions of customers and can involve them into the evaluation, even without the need to let them know. In contrast, most academic research lacks such possibility, focusing instead on offline experiments.

The most common variant of online evaluation is *A/B testing*, i.e., a comparative evaluation of two (or more) recommendation algorithms in a productive

system [68].[37] A/B testing is the most efficient way to evaluate MRS as it allows to measure the system's performance or impact according to the final goals of the system directly in the experiment, using measures such as user retention, click through rate, amount of music streamed, etc. A recent example of A/B testing in an MRS is provided by Spotify in [120], where a multi-armed bandit approach to balance exploration and exploitation, which also provides explanations for recommendations, is proposed and evaluated both offline and online.

4.5.3 User Studies

Evaluation via user studies allows to investigate the user experience of a MRS, including aspects of user engagement [107] and user satisfaction [105]. Pu et al. [142] as well as Knijnenburg et al. [94] propose evaluation frameworks for user-centric evaluation of recommender systems via user studies, which are partly adopted in the MRS domain too. Pu et al.'s framework [142], called *ResQue*, includes aspects of perceived recommendation quality (e.g., attractiveness, novelty, diversity, and perceived accuracy), interface adequacy (e.g., information sufficiency and layout clarity), interaction adequacy (e.g., preference elicitation and revision), as well as perceived usefulness, ease of use, user control, transparency, explicability, and trust. While the authors do not explicitly showcase their framework on a music streaming platform, some results obtained for Youtube[38] and Douban[39]—both platforms heavily used for music consumption—are likely to generalize to dedicated MRS. Knijnenburg et al. [94] propose a different instrument to investigate user experience of recommender systems, which includes aspects such as perceived recommendation quality, perceived system effectiveness, perceived recommendation variety, choice satisfaction, intention to provide feedback, general trust in technology, and system-specific privacy concern, among others.

Conducting user studies to evaluate MRS presents obvious advantages over offline and online experiments because the respective questionnaires are capable of uncovering intrinsic characteristics of user experience to a much deeper degree than the other mentioned strategies. However, such evaluations are rare in the MRS literature as it is difficult to gather a number of participants large enough to draw significant and usable conclusions, due to the required effort on the user side. Existing user studies are typically restricted to a small number of subjects (tens to a few hundreds) and tested approaches or systems. Although the number of user studies has increased [185], conducting such studies on real-world MRS remains time-consuming, expensive, and impractical, particularly for academic researchers.

Consequently, relatively few studies measuring aspects related to user satisfaction have been published, even though their number has been increasing in the

[37] Notwithstanding, there also exist offline variants of A/B testing strategies, e.g. [63].

[38] https://www.youtube.com.

[39] https://www.douban.com.

past years. The study by Celma and Herrera [30] serves as an early example of a subjective evaluation experiment, carried out on a larger scale. Each of the 288 participants provided *liking* (enjoyment of the recommended music) and *familiarity* ratings for 19 tracks recommended by three recommendation approaches. Bogdanov et al. [20] use four subjective measures addressing different aspects of user preference and satisfaction: *liking*, *familiarity* with the recommended tracks, *listening intention* (readiness to listen to the same track again in the future), and "*give-me-more*" (indicating a request for or rejection of more music that is similar to the recommended track). These subjective ratings are analyzed for consistency (a user may like a recommended track, but will not want to listen to it again) and are additionally re-coded into recommendation outcomes: *trusts* (relevant recommendations already known to a user), *hits* (relevant novel tracks), and *fails* (disliked tracks).

Other examples include the recent study by Robinson et al. [147] on perceived diversity in music recommendation lists created by a CF system. The authors investigate the extent to which applying an algorithmic diversification strategy, adopting an intra-list diversity metric, transfers to actual user-perceived diversity. They find a clear difference between diversity preferences in recommendation lists within the user's bounds of music preferences and outside of these bounds.

Another recent study by Jin et al. [76] investigates the extent to which user control over contextual factors that are considered by the recommendation algorithm influences perceived quality, diversity, effectiveness, and cognitive load. In their study, 114 participants are either given no control over the algorithm (realized via Spotify's API) or they could chose a specific context and recommendations are reranked accordingly. The authors find that the users' ability to control whether recommendations are contextualized by mood, weather, and location influences their perception of the MRS. For instance, the ability to consider mood positively affects perceived recommendation quality and diversity.

4.6 How to Deal with Missing and Negative Feedback in Evaluation?

As mentioned throughout this chapter, music recommendation poses several challenges for evaluation. In particular, MRS's are often faced with both implicit feedback (e.g., from the *lean-back* setting) and extreme sparsity of observation. These both contribute to a lack of strong *negative* feedback, which makes standard ranking metrics (derived from precision and recall) difficult to estimate.

The most common approach to cope with sparsity in evaluation is to exploit structure in the content provided by *meta-data*. For example, rather than evaluate a recommender according to its ability to predict interactions between users and songs, the evaluation can be abstracted to measure interactions between users and *artists*. In this view, an item is considered *relevant* if the user interacted with other

items by the same artist. This evaluation obviously over-simplifies the task from the user's perspective, but it can be considerably more stable in practice than a pure item-based evaluation. It is also possible to combine song- and artist-level relevance, as was done in the 2018 RecSys Challenge [33]. Of course, there are other relational structures present in music meta-data that can be leveraged in similar ways. Slaney et al. [164] measured artist, album, and blog co-occurrence as a proxy for (content-based) item similarity, which each gave varying degrees of specificity to the evaluation. Zheleva et al. [192] evaluated playlist generation models by their ability to select songs of appropriate *genres*. While these approaches do not measure utility in the traditional sense, but in the absence of sufficient interaction data, they can at least act as stable proxy metrics.

4.7 How to Design User Interfaces That Match the Use Case and Increase User Experience?

Like most recommender systems domains, music recommendation can benefit from having recommendations presented in a meaningfully organized manner. Many commercial music services present recommendations grouped together by genre, similarity to a popular artist, year of release, etc. Similar experiences are provided by movie recommenders, book recommenders, or general online shopping sites where products may be grouped by "department" (e.g., *kitchen*, *apparel*, *toys*, etc.). The MRS experience differs principally by the high variability of expertise and familiarity with terminology possessed by users. While movies, books, and department stores generally have fairly consistent and familiar "sections" (or shelves), the taxonomies used to categorize and organize music can be both deep and obscure [49, 167]. For example, a casual jazz listener may not understand (or care about) the distinctions between sub-genres like *bebop* or *cool jazz*, but these differences would be obvious to listeners with a bit more familiarity with the genre. This has ramifications for interface design in music recommendation: the groupings used to organize a set of recommendations should adapt to the user's experience and prior knowledge.

Prior knowledge and listening histories are not the only user characteristics that inform MRS (and MRS interface) design. User studies have demonstrated that personality traits (e.g., the Big Five taxonomy [78]) correlate with different preferences for organizational principles in music collections, such as genre, mood, or intended context/activity [58]. This observation motivates the use of hybrid methods which adaptively personalize the combination of recommender algorithms based on each user's listening patterns [51].

Finally, content curation and simple rule-based systems can play a substantial role in music recommendation. Some familiar examples include: removing Christmas music from streams outside the month of December, avoiding the mixing of religious and secular music, filtering music with potentially offensive lyrics,

moderating user-generated content (e.g., podcasts), restricting access to content based on licensing terms and availability, and so on. Streaming radio services additionally may need to comply with regional broadcast regulations, such as avoiding playing multiple songs from the same album within a given window of time. Providing a satisfying user experience under these constraints typically requires a mixture of high-quality meta-data, careful curation and moderation of content, and logical constraints in the recommendation algorithm.

4.8 Which Open Tools and Data Sources can be Used to Build a Music Recommender System?

Many generic cross-domain software tools for recommender systems are available to build MRS. LightFM,[40] Implicit,[41] Surprise,[42] and LibRec[43] provide Python and Java implementations of some popular recommendation algorithms for both implicit and explicit feedback data. Annoy,[44] NMSLIB,[45] and faiss[46] can be used for efficient nearest neighbor search in feature spaces of item and user representations, and are implemented in C++ with Python wrappers.

Here we focus on content annotation tools and data sources particular to music. Specifically, we discuss open-source tools and publicly accessible data that allow bootstrapping MRS, even though some commercial services provide API endpoints to gather music content data and even create recommendations out of the box.

Researchers in MIR have developed *tools* for music audio content analysis and feature extraction, that can be used for music recommendation. Essentia[47] [22] and Librosa[48] [119] provide signal processing algorithms for computation of MIR features and audio representations suitable as inputs for CBF and hybrid approaches (both traditional and deep learning-based). Both libraries provide flexibility of use for fast prototyping in Python. Essentia provides feature extractors implemented in C++ for fast analysis on the large scale. It also includes pre-trained TensorFlow models for auto-tagging and music annotation tasks [8], outputs and latent feature embeddings of which can be useful features for MRS tasks. More audio analysis libraries available to researchers are reviewed in [127].

[40] https://making.lyst.com/lightfm/docs.
[41] https://implicit.readthedocs.io.
[42] http://surpriselib.com.
[43] https://guoguibing.github.io/librec.
[44] https://github.com/spotify/annoy.
[45] https://github.com/nmslib/nmslib.
[46] https://github.com/facebookresearch/faiss.
[47] https://essentia.upf.edu.
[48] https://librosa.org.

The *datasets* presented in Tables 1 and 2 can be used for prototyping MRS. However, their characteristics (provider, origin, release year, composition of user base, type and quality of data, etc.) will bias any recommendation model created from these datasets, likely resulting in problems when translating to the newly built system. Furthermore, it is advisable to check the licenses of the available datasets, as there may be potential legal uncertainty of their usage outside of academic research.

Given the limitations of access to commercial data for researchers and practitioners, there is an initiative to build an open data and open source ecosystem suitable for building recommender systems. MusicBrainz[49] is one of the largest databases of editorial music metadata, including information and relations between millions of artists, recording labels, releases, and particular tracks. It is coupled with ListenBrainz,[50] a database of user listening events, and AcousticBrainz[51] [141], which contains results of automatic music audio analysis and annotation, including high-level music concepts (e.g., genre, mood, instrumentation, key, rhythm). Similar to MusicBrainz, Discogs[52] [21] also provides rich editorial metadata relations and genre/style annotations at the very large scale. All of these data sources are available under open licenses, and they can be used to enrich the data about a particular music collection at hand.

5 Conclusions

To summarize, research and development in music recommender systems has seen a paradigm shift in recent years: away from traditional recommendation tasks such as predicting ratings or impressions, towards more specific *use cases* to satisfy a lean-in or lean-back demand of the user and continuously provide a listening experience. As such, typical applications relevant in music recommendation are, for instance, automatic playlist continuation, cross-modal tasks such as creating a session or playlist based on a textual query, and contextual recommendations based on user intent recognition.

As for *techniques*, we focused on methods that leverage characteristics of the music domain, which distinguishes the music recommendation task from recommendation tasks in other domains. Owed to recent developments, we put a focus on hybrid methods that integrate both co-listening data and content-based information. Furthermore, we reviewed recent research on context-aware, sequential, and psychology-inspired approaches.

Current *challenges* we eventually discuss include considering multi-faceted qualities in recommendation lists, adapting recommendations based on intrinsic

[49] https://www.musicbrainz.org.

[50] https://www.listenbrainz.org.

[51] https://www.acousticbrainz.org.

[52] https://www.discogs.com.

user characteristics, considering fairness and bias, explaining recommendations, evaluating music recommender systems, publicly available datasets, dealing with missing and negative feedback, and designing user interfaces that increase user experience. We are sure that addressing those challenges will yield exciting research results and products in the near future.

Acknowledgments We would like to thank Marius Kaminskas for contributing to the previous version of this chapter, in the second edition of this book.

References

1. M.H. Abdi, G.O. Okeyo, R.W. Mwangi, Matrix factorization techniques for context-aware collaborative filtering recommender systems: a survey. Comput. Inf. Sci. **11**(2), 1–10 (2018)
2. H. Abdollahpouri, G. Adomavicius, R. Burke, I. Guy, D. Jannach, T. Kamishima, J. Krasnodebski, L. Pizzato, Beyond personalization: Research directions in multistakeholder recommendation (2019). arXiv:1905.01986
3. H. Abdollahpouri, R. Burke, B. Mobasher, Recommender systems as multistakeholder environments. in *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, UMAP '17*, New York, NY, 2017 (Association for Computing Machinery, New York, 2017), pp. 347–348
4. H. Abdollahpouri, S. Essinger, Multiple stakeholders in music recommender systems (2017). arXiv:1708.00120
5. H. Abdollahpouri, M. Mansoury, R. Burke, B. Mobasher, The unfairness of popularity bias in recommendation, in *Proceedings of the Workshop on Recommendation in Multi-stakeholder Environments co-located with the 13th ACM Conference on Recommender Systems (RecSys 2019)*, Copenhagen, Denmark, September 20, 2019, ed. by R. Burke, H. Abdollahpouri, E.C. Malthouse, K.P. Thai, Y. Zhang. CEUR Workshop Proceedings, vol. 2440 (CEUR-WS.org, Amsterdam, 2019)
6. H. Abdollahpouri, M. Mansoury, R. Burke, B. Mobasher, The connection between popularity bias, calibration, and fairness in recommendation, in *Fourteenth ACM Conference on Recommender Systems*, RecSys '20, New York, NY, 2020 (Association for Computing Machinery, New York, 2020), pp. 726–731
7. G. Adomavicius, A. Tuzhilin, Context-aware recommender systems, in *Recommender Systems Handbook*, ed. by F. Ricci, L. Rokach, B. Shapira (Springer, New York, 2015), pp. 191–226
8. P. Alonso-Jiménez, D. Bogdanov, J. Pons, X. Serra, Tensorflow audio models in essentia, in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, New York, 2020), pp. 266–270
9. A. Anderson, L. Maystre, I. Anderson, R. Mehrotra, M. Lalmas, Algorithmic effects on the diversity of consumption on spotify, in *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20–24, 2020*, ed. by Y. Huang, I. King, T. Liu, M. van Steen (ACM/IW3C2, New York, 2020), pp. 2155–2165
10. J.R. Anderson, M. Matessa, C. Lebiere, Act-r: a theory of higher level cognition and its relation to visual attention. Human-Computer Interact. **12**(4), 439–462 (1997)
11. I. Andjelkovic, D. Parra, J. O'Donovan, Moodplay: Interactive mood-based music discovery and recommendation, in *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization*, UMAP '16, New York, NY (ACM, New York, 2016), pp. 275–279
12. D. Ayata, Y. Yaslan, M.E. Kamasak, Emotion based music recommendation system using wearable physiological sensors. IEEE Trans. Consum. Electron. **64**(2), 196–203 (2018)

13. R. Baeza-Yates, B.A. Ribeiro-Neto, *Modern Information Retrieval - The Concepts and Technology Behind Search*, 2nd edn. (Pearson Education Ltd., Harlow, 2011)
14. L. Baltrunas, B. Ludwig, F. Ricci, Matrix factorization techniques for context aware recommendation, in *Proceedings of the 2011 ACM Conference on Recommender Systems, RecSys 2011, Chicago, IL, October 23–27, 2011*, ed. by B. Mobasher, R.D. Burke, D. Jannach, G. Adomavicius, pp. 301–304 (ACM, New York, 2011)
15. L. Baltrunas, F. Ricci, Context-based splitting of item ratings in collaborative filtering, in *Proceedings of the 2009 ACM Conference on Recommender Systems, RecSys 2009, New York, NY, October 23–25, 2009*, ed. by L.D. Bergman, A. Tuzhilin, R.D. Burke, A. Felfernig, L. Schmidt-Thieme (ACM, New York, 2009), pp. 245–248
16. C. Bauer, A. Novotny, A consolidated view of context for intelligent systems. J. Ambient Intell. Smart Environ. **9**(4), 377–393 (2017)
17. C. Bauer, M. Schedl, Global and country-specific mainstreaminess measures: definitions, analysis, and usage for improving personalized music recommendation systems. PLoS One **14**(6), 1–36 (2019)
18. T. Bertin-Mahieux, D.P. Ellis, B. Whitman, P. Lamere, The million song dataset, in *Proceedings of the 12th International Society for Music Information Retrieval Conference*, Miami, October 24–28 2011, pp. 591–596
19. A. Beutel, P. Covington, S. Jain, C. Xu, J. Li, V. Gatto, E.H. Chi, Latent cross: Making use of context in recurrent recommender systems. In ed. by Y. Chang, C. Zhai, Y. Liu, Y. Maarek, *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5–9, 2018* (ACM, New York, 2018), pp. 46–54
20. D. Bogdanov, M. Haro, F. Fuhrmann, A. Xambó, E. Gómez, P. Herrera, Semantic audio content-based music recommendation and visualization based on user preference examples. Inf. Process. Manag. **49**(1), 13–33 (2013)
21. D. Bogdanov, P. Herrera, Taking advantage of editorial metadata to recommend music, in *Int. Symp. on Computer Music Modeling and Retrieval (CMMR'12)*, 2012
22. D. Bogdanov, N. Wack, E. Gómez Gutiérrez, S. Gulati, H. Boyer, O. Mayor, G. Roma Trepat, J. Salamon, J. R. Zapata González, X. Serra, et al., Essentia: an audio analysis library for music information retrieval, in *Britto A, Gouyon F, Dixon S, editors. 14th Conference of the International Society for Music Information Retrieval (ISMIR); 2013 Nov 4–8; Curitiba, Brazil.[place unknown]: ISMIR; 2013. p. 493–498.* International Society for Music Information Retrieval (ISMIR), 2013.
23. B. Brost, R. Mehrotra, T. Jehan, The music streaming sessions dataset, in L. Liu, R.W. White, A. Mantrach, F. Silvestri, J.J. McAuley, R. Baeza-Yates, L. Zia, editors, *The World Wide Web Conference, WWW 2019, San Francisco, CA, May 13–17, 2019* (ACM, New York, 2019), pp. 2594–2600
24. Burke, R., Multisided fairness for recommendation (2017). CoRR abs/1707.00093. arXiv
25. R.D. Burke, Hybrid recommender systems: Survey and experiments. User Model. User Adapt. Interact. **12**(4), 331–370 (2002)
26. R.D. Burke, M. Mansoury, N. Sonboli, Experimentation with fairness-aware recommendation using librec-auto: Hands-on tutorial, in *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, p. 700, New York, NY 2020. Association for Computing Machinery.
27. P. Castells, N.J. Hurley, S. Vargas, Novelty and diversity in recommender systems, in *Recommender Systems Handbook* (Springer, Boston, MA, 2015), pp. 881–918
28. Ò. Celma, *Music Recommendation and Discovery – The Long Tail, Long Fail, and Long Play in the Digital Music Space* (Springer, Berlin, 2010)
29. O. Celma, The exploit-explore dilemma in music recommendation. In *Proceedings of the 10th ACM Conference on Recommender Systems* (2016), pp. 377–377
30. O. Celma, P. Herrera, A new approach to evaluating novel recommendations, in *ACM Conference on Recommender Systems (RecSys'08)* (2008), pp. 179–186

31. S. Chang, F.M. Harper, L.G. Terveen, Crowd-based personalized natural language explanations for recommendations, in *Proc. ACM Conf. on Recommender Systems, RecSys '16*, pp. 175–182 (ACM, New York, 2016)
32. S. Chang, S. Lee, K. Lee, Sequential skip prediction with few-shot in streamed music contents. CoRR abs/1901.08203, 2019.
33. C.-W. Chen, P. Lamere, M. Schedl, and H. Zamani. Recsys challenge 2018: Automatic music playlist continuation. In *Proceedings of the 12th ACM Conference on Recommender Systems*, RecSys '18, page 527–528, New York, NY, USA, 2018. Association for Computing Machinery.
34. R. Chen, Q. Hua, Y. Chang, B. Wang, L. Zhang, X. Kong, A survey of collaborative filtering-based recommender systems: from traditional methods to hybrid methods based on social networks. IEEE Access **6**, 64301–64320 (2018)
35. Z. Cheng, J. Shen, On effective location-aware music recommendation. ACM Trans. Inf. Syst. (TOIS) **34**(2), 1–32 (2016)
36. S.J. Cunningham, Interacting with personal music collections. in *Information in Contemporary Society*, 2019 (Springer International Publishing, Cham, 2019), pp. 526–536
37. S.J. Cunningham, D. Bainbridge, A. Bainbridge, Exploring personal music collection behavior, in ed. by S. Choemprayong, F. Crestani, S.J. Cunningham, *Digital Libraries: Data, Information, and Knowledge for Digital Lives* (Springer International Publishing, Cham, 2017), pp. 295–306
38. S.J. Cunningham, D. Bainbridge, A. Falconer, 'More of an art than a science': supporting the creation of playlists and mixes, in *ISMIR 2006, 7th International Conference on Music Information Retrieval, Victoria, 8–12 October 2006, Proceedings* (2006), pp 240–245.
39. S.J. Cunningham, D. Bainbridge, D. Mckay, Finding new music: a diary study of everyday encounters with novel songs, in *Proceedings of the 8th International Conference on Music Information Retrieval*, pp. 83–88, Vienna, September 23–27 (2007)
40. S.J. Cunningham, J.S. Downie, D. Bainbridge, The pain, the pain: modelling music information behavior and the songs we hate, in *ISMIR 2005, 6th International Conference on Music Information Retrieval, London, 11–15 September 2005, Proceedings* (2005), pp. 474–477
41. Y. Deldjoo, M. Schedl, P. Cremonesi, G. Pasi, Recommender systems leveraging multimedia content. ACM Computing Surv. **53**(5) (2020)
42. Y. Deldjoo, M. Schedl, P. Knees, Content-driven music recommendation: evolution, state of the art, and challenges (2021). Preprint. arXiv
43. S. Deng, D. Wang, X. Li, G. Xu, Exploring user emotion in microblogs for music recommendation. Expert Syst. Appl. **42**(23), 9284–9293 (2015)
44. G. Dror, N. Koenigstein, Y. Koren, M. Weimer, The Yahoo! Music Dataset and KDD-Cup'11. J. Mach. Learn. Res. Proc. KDD-Cup 2011 Compet. **18**, 3–18 (2012)
45. P.G. Dunn, B. de Ruyter, D.G. Bouwhuis, Toward a better understanding of the relation between music preference, listening behavior, and personality. Psychol. Music **40**(4), 411–428 (2012)
46. T. Eerola, J. Vuoskoski, A comparison of the discrete and dimensional models of emotion in music. Psychol. Music **39**(1), 18–49 (2011)
47. H. Eghbal-zadeh, B. Lehner, M. Schedl, G. Widmer, I-vectors for timbre-based music similarity and music artist classification, in *Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR 2015, Málaga, October 26–30, 2015*, ed. by M. Müller, F. Wiering (2015), pp. 554–560
48. M.D. Ekstrand, M. Tian, I.M. Azpiazu, J.D. Ekstrand, O. Anuyah, D. McNeill, M.S. Pera, All the cool kids, how do they fit in?: Popularity and demographic biases in recommender evaluation and effectiveness, in *Conference on Fairness, Accountability and Transparency, FAT 2018, 23–24 February 2018, New York, NY*, ed. by S.A. Friedler, C. Wilson. Proceedings of Machine Learning Research, vol. 81 (PMLR, 2018), pp. 172–186
49. F. Fabbri, A theory of musical genres: two applications. Popul. Mus. Perspect. **1**, 52–81 (1982)

50. I. Fernández-Tobías, M. Braunhofer, M. Elahi, F. Ricci, I. Cantador, Alleviating the new user problem in collaborative filtering by exploiting personality information. User Model. User-Adapt. Interact. **26**(2–3), 221–255 (2016)
51. A. Ferraro, D. Bogdanov, K. Choi, X. Serra, Using offline metrics and user behavior analysis to combine multiple systems for music recommendation. in *Proceedings of the RecSys 2018 Workshop on Offline Evaluation of Recommender Systems (REVEAL)* (2018), pp. 6
52. A. Ferraro, Y. Kim, S. Lee, B. Kim, N. Jo, S. Lim, S. Lim, J. Jang, S. Kim, X. Serra, et al., Melon playlist dataset: a public dataset for audio-based playlist generation and music tagging. in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, New York, 2021), pp. 536–540
53. B. Ferwerda, M. Graus, A. Vall, M. Tkalčič, M. Schedl, The influence of users' personality traits on satisfaction and attractiveness of diversified recommendation lists. in *4th Workshop on Emotions and Personality in Personalized Systems (EMPIRE) 2016* (2016), p. 43
54. B. Ferwerda, M. Schedl, M. Tkalčič, Personality & emotional states: understanding users' music listening needs, in *Extended Proceedings of the 23rd International Conference on User Modeling, Adaptation and Personalization (UMAP)*, Dublin, June–July 2015
55. B. Ferwerda, M. Tkalčič, M. Schedl, Personality traits and music genre preferences: How music taste varies over age groups, in *Proceedings of the 1st Workshop on Temporal Reasoning in Recommender Systems (RecTemp) at the 11th ACM Conference on Recommender Systems, Como, August 31, 2017*, 2017
56. B. Ferwerda, M. Tkalcic, M. Schedl, Personality traits and music genres: What do people prefer to listen to? in *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*, UMAP '17, New York, NY, (ACM, New York, 2017), pp. 285–288
57. B. Ferwerda, E. Yang, M. Schedl, M. Tkalčič, Personality traits predict music taxonomy preferences, in *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems* (ACM, New York, 2015), pp. 2241–2246
58. B. Ferwerda, E. Yang, M. Schedl, M. Tkalcic, Personality and taxonomy preferences, and the influence of category choice on the user experience for music streaming services. Multim. Tools Appl. **78**(14), 20157–20190 (2019)
59. B. Fields, Contextualize your listening: the playlist as recommendation engine. PhD thesis, Department of Computing Goldsmiths, University of London, 2011
60. K.R. Fricke, D.M. Greenberg, P.J. Rentfrow, P.Y. Herzberg, Computer-based music feature analysis mirrors human perception and can be used to measure individual music preference. J. Res. Personal. **75**, 94–102 (2018)
61. G. Friedrich, M. Zanker, A taxonomy for generating explanations in recommender systems. AI Mag. **32**(3), 90–98 (2011)
62. A. Gautam, P. Chaudhary, K. Sindhwani, P. Bedi, CBCARS: content boosted context-aware recommendations using tensor factorization, in *2016 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2016, Jaipur, September 21–24, 2016* (IEEE, New York, 2016), pp. 75–81
63. A. Gilotte, C. Calauzènes, T. Nedelec, A. Abraham, S. Dollé, Offline a/b testing for recommender systems, in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, WSDM '18, New York, NY (Association for Computing Machinery, New York, 2018), pp. 198–206
64. M. Goto, R.B. Dannenberg, Music interfaces based on automatic music signal analysis: new ways to create and listen to music. IEEE Signal Process. Mag. **36**(1), 74–81 (2019)
65. M. Goto, K. Yoshii, H. Fujihara, M. Mauch, T. Nakano, Songle: a web service for active music listening improved by user contributions, in *Proceedings of the 12th International Society for Music Information Retrieval Conference*, pp. 311–316, Miami, October. 2011. ISMIR.
66. S.J. Green, P. Lamere, J. Alexander, F. Maillet, S. Kirk, J. Holt, J. Bourque, X. Mak, Generating transparent, steerable recommendations from textual descriptions of items, in *Proceedings of the 2009 ACM Conference on Recommender Systems, RecSys 2009*, New York, NY, October 23–25, 2009, ed. by L.D. Bergman, A. Tuzhilin, R.D. Burke, A. Felfernig, L. Schmidt-Thieme (ACM, New York, 2009), pp. 281–284

67. S.J. Green, P. Lamere, J. Alexander, F. Maillet, S. Kirk, J. Holt, J. Bourque, X.-W. Mak, Generating transparent, steerable recommendations from textual descriptions of items, in *Proc. ACM Conf. on Recommender Systems, RecSys '09* (ACM, New York, 2009), pp. 281–284
68. A. Gunawardana, G. Shani, Evaluating recommender systems, in *Recommender Systems Handbook*, ed. by F. Ricci, L. Rokach, B. Shapira (Springer, New York, 2015), pp. 265–308
69. C. Hansen, C. Hansen, S. Alstrup, J.G. Simonsen, C. Lioma, Modelling sequential music track skips using a multi-rnn approach. CoRR abs/1903.08408, 2019
70. D. Hauger, M. Schedl, A. Košir, M. Tkalčič, 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, November 2013
71. J.L. Herlocker, J.A. Konstan, L.G. Terveen, J.T. Riedl, Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. **22**(1), 5–53 (2004)
72. K. Hevner, Expression in music: a discussion of experimental studies and theories. Psychol. Rev. **42**, 186–204 (1935)
73. Y. Hu, Y. Koren, C. Volinsky, Collaborative filtering for implicit feedback datasets, in *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008)*, December 15–19, 2008, Pisa (IEEE Computer Society, Washington, 2008), pp. 263–272
74. Q. Huang, A. Jansen, L. Zhang, D.P.W. Ellis, R.A. Saurous, J.R. Anderson, Large-scale weakly-supervised content embeddings for music recommendation and tagging, in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020*, Barcelona, May 4–8, 2020 (IEEE, New York, 2020), pp. 8364–8368
75. D. Jannach, M. Zanker, A. Felfernig, G. Friedrich, *Recommender Systems - An Introduction* (Cambridge University Press, Cambridge, 2010)
76. Y. Jin, N.N. Htun, N. Tintarev, K. Verbert, Contextplay: Evaluating user control for context-aware music recommendation, in *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization, UMAP 2019*, Larnaca, Cyprus, June 9–12, 2019, ed. by G.A. Papadopoulos, G. Samaras, S. Weibelzahl, D. Jannach, O.C. Santos (ACM, New York, 2019)
77. T. Joachims, Optimizing search engines using clickthrough data, in *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2002), pp. 133–142
78. O.P. John, E.M. Donahue, R.L. Kentle, The big five inventory—versions 4a and 54 (1991)
79. P. Juslin, P. Laukka, Expression, perception, and induction of musical emotions: a review and a questionnaire study of everyday listening. J. New Music Res. **33**(2), 217–238 (2004)
80. I. Kamehkhosh, G. Bonnin, D. Jannach, Effects of recommendations on the playlist creation behavior of users, in *User Modeling and User-Adapted Interaction*, 2019
81. I. Kamehkhosh, D. Jannach, G. Bonnin, How automated recommendations affect the playlist creation behavior of users, in *Joint Proceedings of the ACM IUI 2018 Workshops co-located with the 23rd ACM Conference on Intelligent User Interfaces (ACM IUI 2018), Tokyo, March 11, 2018*, ed. by A. Said, T. Komatsu. CEUR Workshop Proceedings, vol. 2068 (CEUR-WS.org, Amsterdam, 2018)
82. M. Kaminskas, D. Bridge, Diversity, serendipity, novelty, and coverage: a survey and empirical analysis of beyond-accuracy objectives in recommender systems. ACM Trans. Interact. Intell. Syst. **7**(1), 2:1–2:42 (2017)
83. M. Kaminskas, F. Ricci, Contextual music information retrieval and recommendation: state of the art and challenges. Comput. Sci. Rev. **6**, 89–119 (2012)
84. M. Kaminskas, F. Ricci, M. Schedl, Location-aware music recommendation using auto-tagging and hybrid matching, in *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys 2013)*, Hong Kong, October 2013
85. A. Karatzoglou, X. Amatriain, L. Baltrunas, N. Oliver, Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering, in *Proceedings of the 2010 ACM Conference on Recommender Systems, RecSys 2010*, Barcelona, September 26–30, 2010, ed. by X. Amatriain, M. Torrens, P. Resnick, M. Zanker (eds.) (ACM, New York, 2010), pp. 79–86

86. E. Karydi, K.G. Margaritis, Parallel and distributed collaborative filtering: a survey. ACM Comput. Surv. **49**(2), 37:1–37:41 (2016)
87. Y. Kjus, Musical exploration via streaming services: The norwegian experience. Popul. Commun. **14**(3), 127–136 (2016)
88. P. Knees, A proposal for a neutral music recommender system, in , *Proceedings of the 1st Workshop on Designing Human-Centric Music Information Research Systems*, ed. by M. Miron (2019), pp. 4–7
89. P. Knees, M. Hübler, Towards uncovering dataset biases: investigating record label diversity in music playlists, in *Proceedings of the 1st Workshop on Designing Human-Centric Music Information Research Systems*, ed. by M. Miron (2019), pp. 19–22
90. P. Knees, M. Schedl, A survey of music similarity and recommendation from music context data. ACM Trans. Multimedia Comput. Commun. Appl. **10**(1), 2:1–2:21 (2013)
91. P. Knees, M. Schedl, *Music Similarity and Retrieval - An Introduction to Audio- and Web-based Strategies*, vol. 36. The Information Retrieval Series (Springer, New York, 2016)
92. P. Knees, M. Schedl, B. Ferwerda, A. Laplante, User awareness in music recommender systems, in *Personalized Human-Computer Interaction*, ed. by M. Augstein, E. Herder, W. Wörndl (DeGruyter, Berlin, Boston, 2019), pp. 223–252
93. P. Knees, M. Schedl, M. Goto, Intelligent user interfaces for music discovery. Trans. Int. Soc. Music Inf. Retriev. **3**, 165—179 (2020)
94. B.P. Knijnenburg, M.C. Willemsen, Z. Gantner, H. Soncu, C. Newell, Explaining the user experience of recommender systems. User Model. User Adapt. Interact. **22**(4–5), 441–504 (2012)
95. V.J. Konecni, Social interaction and musical preference, in *The Psychology of Music* (Academic, New York, 1982), pp. 497–516
96. Y. Koren, R.M. Bell, Advances in collaborative filtering, in *Recommender Systems Handbook*, ed. by F. Ricci, L. Rokach, B. Shapira (Springer, New York, 2015), pp. 77–118
97. D. Kowald, E. Lex, M. Schedl, Utilizing human memory processes to model genre preferences for personalized music recommendations (2020). CoRR abs/2003.10699
98. D. Kowald, M. Schedl, E. Lex, The unfairness of popularity bias in music recommendation: a reproducibility study, in *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020*, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part II, ed. by J.M. Jose, E. Yilmaz, J. Magalhães, P. Castells, N. Ferro, M. J. Silva, F. Martins. Lecture Notes in Computer Science, vol. 12036 (Springer, New York, 2020), pp. 35–42
99. W. Krichene, S. Rendle, On sampled metrics for item recommendation. in *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event*, CA, August 23–27, 2020, ed. by R. Gupta, Y. Liu, J. Tang, B.A. Prakash (ACM, New York, 2020), pp. 1748–1757
100. F.-F. Kuo, M.-K. Shan, S.-Y. Lee, Background music recommendation for video based on multimodal latent semantic analysis, in *2013 IEEE International Conference on Multimedia and Expo (ICME)* (IEEE, New York, 2013), pp. 1–6
101. A. Laplante, Everyday life music information-seeking behaviour of young adults: An exploratory study. Doctoral dissertation, 2008
102. A. Laplante, Improving music recommender systems: What we can learn from research on music tastes? in *15th International Society for Music Information Retrieval Conference*, Taipei, Taiwan, October 2014
103. A. Laplante, J.S. Downie, Everyday life music information-seeking behaviour of young adults, in *Proceedings of the 7th International Conference on Music Information Retrieval*, Victoria (BC), October 8–12, 2006
104. J.H. Lee, How similar is too similar?: Exploring users' perceptions of similarity in playlist evaluation, in *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011*, Miami, FL, October 24–28, 2011, ed. by A. Klapuri, C. Leider (University of Miami, Miami, 2011), pp. 109–114

105. J.H. Lee, H. Cho, Y.-S. Kim, Users' music information needs and behaviors: Design implications for music information retrieval systems. J. Assoc. Inf. Sci. Technol. **67**(6), 1301–1330 (2016)
106. J.H. Lee, R. Wishkoski, L. Aase, P. Meas, C. Hubbles, Understanding users of cloud music services: selection factors, management and access behavior, and perceptions. J. Assoc. Inf. Sci. Technol. **68**(5), 1186–1200 (2017)
107. J. Lehmann, M. Lalmas, E. Yom-Tov, G. Dupret, Models of user engagement, in *User Modeling, Adaptation, and Personalization - 20th International Conference, UMAP 2012*, Montreal, July 16–20, 2012. Proceedings, ed. by J. Masthoff, B. Mobasher, M.C. Desmarais, R. Nkambou. Lecture Notes in Computer Science, , vol. 7379, pp. 164–175 (Springer, New York, 2012)
108. E. Lex, D. Kowald, P. Seitlinger, T.N.T. Tran, A. Felfernig, M. Schedl, Psychology-informed recommender systems, in *Foundations and Trends in Information Retrieval*, 2021
109. Q. Lin, Y. Niu, Y. Zhu, H. Lu, K.Z. Mushonga, Z. Niu, Heterogeneous knowledge-based attentive neural networks for short-term music recommendations. IEEE Access **6**, 58990–59000 (2018)
110. Y.-T. Lin, T.-H. Tsai, M.-C. Hu, W.-H. Cheng, J.-L. Wu, Semantic based background music recommendation for home videos, in *International Conference on Multimedia Modeling* (Springer, New York, 2014), pp. 283–290
111. A.J. Lonsdale, A.C. North, Why do we listen to music? A uses and gratifications analysis. Br. J. Psychol. **102**(1), 108–134 (2011)
112. C.-C. Lu, V.S. Tseng, A novel method for personalized music recommendation. Expert Syst. Appl. **36**(6), 10035–10044 (2009)
113. F. Lu, N. Tintarev, A diversity adjusting strategy with personality for music recommendation, in *Proceedings of the 5th Joint Workshop on Interfaces and Human Decision Making for Recommender Systems, co-located with ACM Conference on Recommender Systems (RecSys 2018)*, October 2018, pp. 7–14
114. B. McFee, L. Barrington, G. Lanckriet, Learning content similarity for music recommendation. IEEE Trans. Audio Speech Lang. Process. **20**(8), 2207–2218 (2012)
115. B. McFee, T. Bertin-Mahieux, D. Ellis, and G. Lanckriet. The million song dataset challenge. In *Proc. of the 4th International Workshop on Advances in Music Information Research (AdMIRe)*, April 2012.
116. B. McFee, G. Lanckriet, The natural language of playlists, in *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, Miami, FL, 2011
117. B. McFee, G. Lanckriet, Hypergraph models of playlist dialects, in *Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR 2012)*, Porto, October 2012
118. B. McFee, G.R.G. Lanckriet, Learning multi-modal similarity. J. Mach. Learn. Res. **12**, 491–523 (2011)
119. B. McFee, C. Raffel, D. Liang, D.P. Ellis, M. McVicar, E. Battenberg, O. Nieto, librosa: audio and music signal analysis in python, in *Proceedings of the 14th Python in Science Conference*, vol. 8 (2015), pp. 18–25
120. J. McInerney, B. Lacker, S. Hansen, K. Higley, H. Bouchard, A. Gruson, R. Mehrotra, Explore, exploit, and explain: Personalizing explainable recommendations with bandits, in *Proceedings of the 12th ACM Conference on Recommender Systems*, RecSys '18, New York, NY, (Association for Computing Machinery, New York, 2018), pp. 31–39
121. S. McNee, J. Riedl, J. Konstan, Being accurate is not enough: how accuracy metrics have hurt recommender systems, in *CHI'06 Extended Abstracts on Human Factors in Computing Systems* (2006), p. 1101
122. R. Mehrotra, J. McInerney, H. Bouchard, M. Lalmas, F. Diaz, Towards a fair marketplace: Counterfactual evaluation of the trade-off between relevance, fairness & satisfaction in recommendation systems, in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18 (Association for Computing Machinery, New York, NY, 2018), pp. 2243–2251

123. A.B. Melchiorre, M. Schedl, Personality correlates of music audio preferences for modelling music listeners, in *Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '20 (Association for Computing Machinery, New York, NY, 2020), pp. 313–317
124. A.B. Melchiorre, E. Zangerle, M. Schedl, Personality bias of music recommendation algorithms, in *Fourteenth ACM Conference on Recommender Systems, RecSys '20* (Association for Computing Machinery, New York, NY, 2020), pp. 533–538
125. M. Millecamp, N.N. Htun, C. Conati, K. Verbert, To explain or not to explain: the effects of personal characteristics when explaining music recommendations, in *Proceedings of the 24th International Conference on Intelligent User Interfaces, IUI 2019*, Marina del Ray, CA, March 17–20, 2019, ed. by W. Fu, S. Pan, O. Brdiczka, P. Chau, G. Calvary (ACM, New York, 2019), pp. 397–407
126. M. Millecamp, N.N. Htun, Y. Jin, K. Verbert, Controlling spotify recommendations: Effects of personal characteristics on music recommender user interfaces, in *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization, UMAP '18* (Association for Computing Machinery, New York, NY, 2018), pp. 101–109
127. D. Moffat, D. Ronan, J.D. Reiss, An evaluation of audio feature extraction toolboxes, in *18th International Conference on Digital Audio Effects (DAFx-15)* (2015), p. 7
128. M. Müller, *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications* (Springer, New York, 2015)
129. C. Musto, F. Narducci, P. Lops, M. De Gemmis, G. Semeraro, ExpLOD: a framework for explaining recommendations based on the LOD cloud, in *Proc. ACM Conf. on Recommender Systems, RecSys '16* (ACM, New York, 2016), pp. 151–154
130. T. Nakano, M. Goto, LyricListPlayer: a consecutive-query-by-playback interface for retrieving similar word sequences from different song lyrics, in *Proceedings of the 13th Sound and Music Computing Conference (SMC2016)*, Hamburg, August 2016, Zenodo
131. A.C. North, D.J. Hargreaves, Subjective complexity, familiarity, and liking for popular music. Psychomusicol. Music Mind Brain **14**(1–2), 77–93 (1995)
132. A.C. North, D.J. Hargreaves, Situational influences on reported musical preference. Psychomusicol. J. Res. Music Cogn. **15**(1–2), 30 (1996)
133. S. Oramas, O. Nieto, M. Sordo, X. Serra, A deep multimodal approach for cold-start music recommendation, in *Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems, DLRS@RecSys 2017, Como, August 27, 2017*, ed. by B. Hidasi, A. Karatzoglou, O.S. Shalom, S. Dieleman, B. Shapira, D. Tikk (ACM, New York, 2017), pp. 32–37
134. S. Oramas, V.C. Ostuni, T.D. Noia, X. Serra, E.D. Sciascio, Sound and music recommendation with knowledge graphs. ACM Trans. Intell. Syst. Technol. **8**(2), 1–2 (2016)
135. E. Pampalk, M. Goto, Musicsun: a new approach to artist recommendation, in *Proceedings of the 8th International Conference on Music Information Retrieval, ISMIR 2007*, Vienna, September 23–27, 2007, ed. by S. Dixon, D. Bainbridge, R. Typke (Austrian Computer Society, Vienna, 2007), pp. 101–104
136. P. Papreja, H. Venkateswara, S. Panchanathan, Representation, exploration and recommendation of music playlists (2019). Preprint. arXiv:1907.01098
137. D. Parra, X. Amatriain, Walk the talk, in *International Conference on User Modeling, Adaptation, and Personalization* (Springer, New York, 2011), pp. 255–268
138. C.S. Pereira, J. Teixeira, P. Figueiredo, J. Xavier, S.L. Castro, E. Brattico, Music and emotions in the brain: familiarity matters. PLOS One **6**(11), 1–9 (2011)
139. M. Pichl, E. Zangerle, G. Specht, Towards a Context-Aware Music Recommendation Approach: What is Hidden in the Playlist Name? in *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, November 2015, Atlantic City, NJ (IEEE, New York, 2015), pp. 1360–1365
140. A. Poddar, E. Zangerle, Y.-H. Yang, #nowplaying-rs: A new benchmark dataset for building context-aware music recommender systems, in *Proceedings of the 15th Sound & Music Computing Conference*, Limassol, Cyprus, 2018. Code at https://github.com/asmitapoddar/nowplaying-RS-Music-Reco-FM

141. A. Porter, D. Bogdanov, R. Kaye, R. Tsukanov, X. Serra, Acousticbrainz: a community platform for gathering music information obtained from audio, in *International Society for Music Information Retrieval Conference (ISMIR'15)*, 2015
142. P. Pu, L. Chen, R. Hu, A user-centric evaluation framework for recommender systems, in *Proceedings of the 2011 ACM Conference on Recommender Systems, RecSys 2011*, Chicago, IL, October 23–27, 2011, ed. by B. Mobasher, R.D. Burke, D. Jannach, G. Adomavicius (ACM, New York, 2011), pp. 157–164
143. M. Quadrana, P. Cremonesi, D. Jannach, Sequence-aware recommender systems. ACM Comput. Surv. **51**(4), 66:1–66:36 (2018)
144. P.J. Rentfrow, S.D. Gosling, The do re mi's of everyday life: The structure and personality correlates of music preferences. J. Personal. Soc. Psychol. **84**(6), 1236–1256 (2003)
145. P.J. Rentfrow, S.D. Gosling, The content and validity of music-genre stereotypes among college students. Psychol. Music **35**(2), 306–326 (2007)
146. M. T. Ribeiro, S. Singh, and C. Guestrin. "Why Should I Trust You?". In *Proc. Intl. Conf. on Knowledge Discovery and Data Mining* (ACM, New York, 2016), pp. 1135–1144
147. K. Robinson, D. Brown, M. Schedl, User insights on diversity in music recommendation lists, in *Proceedings of the 21st International Society for Music Information Retrieval Conference (ISMIR 2020)*, Virtual, October 2020
148. N. Sachdeva, K. Gupta, V. Pudi, Attentive neural architecture incorporating song features for music recommendation, in *Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018*, Vancouver, BC, Canada, October 2–7, 2018, ed. by S. Pera, M.D. Ekstrand, X. Amatriain, J. O'Donovan (ACM, New York, 2018), pp. 417–421
149. T. Schäfer, P. Sedlmeier, C. Städtler, D. Huron, The psychological functions of music listening. Front. Psychol. **4**(511), 1–34 (2013)
150. M. Schedl, Leveraging microblogs for spatiotemporal music information retrieval, in *Proceedings of the 35th European Conference on Information Retrieval (ECIR 2013)*, Moscow, March 24–27 (2013)
151. M. Schedl, The lfm-1b dataset for music retrieval and recommendation, in *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval, ICMR 2016*, New York, New York, June 6–9, 2016, ed. by J.R. Kender, J.R. Smith, J. Luo, S. Boll, W.H. Hsu (ACM, New York, 2016), pp. 103–110
152. M. Schedl, Deep learning in music recommendation systems. Front. Appl. Math. Stat. **5**, 44 (2019)
153. M. Schedl, C. Bauer, W. Reisinger, D. Kowald, E. Lex, Listener modeling and context-aware music recommendation based on country archetypes. Front. Artif. Intell. **3**, 508725 (2020)
154. M. Schedl, B. Ferwerda, Large-scale analysis of group-specific music genre taste from collaborative tags, in *19th IEEE International Symposium on Multimedia, ISM 2017*, Taichung, December 11–13, 2017 (IEEE Computer Society, New York, 2017), pp. 479–482
155. M. Schedl, E. Gómez, E.S. Trent, M. Tkalcic, H. Eghbal-Zadeh, A. Martorell, On the interrelation between listener characteristics and the perception of emotions in classical orchestra music. IEEE Trans. Affect. Comput. **9**(4), 507–525 (2018)
156. M. Schedl, D. Hauger, Tailoring music recommendations to users by considering diversity, mainstreaminess, and novelty, in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Santiago, August 9–13, 2015, ed. by R. Baeza-Yates, M. Lalmas, A. Moffat, B.A. Ribeiro-Neto (ACM, New York, 2015), pp. 947–950
157. M. Schedl, D. Hauger, K. Farrahi, M. Tkalcic, On the influence of user characteristics on music recommendation algorithms, in *Advances in Information Retrieval - 37th European Conference on IR Research, ECIR 2015*, , Vienna, Austria, March 29 - April 2, 2015. Proceedings, ed. by A. Hanbury, G. Kazai, A. Rauber, N. Fuhr. Lecture Notes in Computer Science, vol. 9022 (2015), pp. 339–345
158. M. Schedl, P. Knees, F. Gouyon, New paths in music recommender systems research, in *Proceedings of the Eleventh ACM Conference on Recommender Systems, RecSys 2017*, Como, August 27–31, 2017, ed. by P. Cremonesi, F. Ricci, S. Berkovsky, A. Tuzhilin (ACM, New York, 2017), pp. 392–393

159. M. Schedl, P. Knees, B. McFee, D. Bogdanov, M. Kaminskas, Music recommender systems, in *Recommender Systems Handbook*, 2nd edn., ed. by F. Ricci, L. Rokach, B. Shapira. (Springer, New York, 2015), pp. 453–492
160. M. Schedl, M. Tkalcic, Genre-based analysis of social media data on music listening behavior: are fans of classical music really averse to social media? in *Proceedings of the First International Workshop on Internet-Scale Multimedia Management, WISMM '14*, , Orlando, FL, November 7, 2014, ed. by R. Zimmermann, Y. Yu (ACM, New York, 2014), pp. 9–13
161. M. Schedl, H. Zamani, C. Chen, Y. Deldjoo, M. Elahi, Current challenges and visions in music recommender systems research. Int. J. Multim. Inf. Retr. **7**(2), 95–116 (2018)
162. G. Shani, A. Gunawardana, Evaluating recommender systems, in *Recommender Systems Handbook* (Springer, New York, 2009), pp. 257–298
163. G. Shani, D. Heckerman, R.I. Brafman, An MDP-based recommender system. J. Mach. Learn. Res. **6**, 1265–1295 (2005)
164. M. Slaney, K. Weinberger, W. White, Learning a metric for music similarity, in *Int. Symp. on Music Information Retrieval (ISMIR'08)* (2008), pp. 313–318
165. J. Smith, D. Weeks, M. Jacob, J. Freeman, B. Magerko, Towards a hybrid recommendation system for a sound library, in *IUI Workshops* (2019)
166. B. Smyth, P. McClave, Similarity vs. diversity, in *Case-Based Reasoning Research and Development, 4th International Conference on Case-Based Reasoning, ICCBR 2001*, Vancouver, BC, Canada, July 30 - August 2, 2001, Proceedings, ed. by D.W. Aha, I.D. Watson. Lecture Notes in Computer Science, vol. 2080 (Springer, New York, 2001), pp. 347–361
167. M. Sordo, O. Celma, M. Blech, E. Guaus, The quest for musical genres: Do the experts and the wisdom of crowds agree? in *Int. Conf. of Music Information Retrieval (ISMIR'08)* (2008), pp. 255–260
168. L. Spinelli, J. Lau, L. Pritchard, J.H. Lee, Influences on the social practices surrounding commercial music services: a model for rich interactions, in *Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR)*, Paris, 2018
169. H. Steck, Calibrated recommendations, in *Proceedings of the 12th ACM Conference on Recommender Systems, RecSys '18* (Association for Computing Machinery, New York, NY, 2018), pp. 154–162
170. A. Swaminathan, T. Joachims, Counterfactual risk minimization: learning from logged bandit feedback, in *International Conference on Machine Learning* (2015), pp. 814–823
171. M. Tiemann, S. Pauws, Towards ensemble learning for hybrid music recommendation, in *ACM Conf. on Recommender Systems (RecSys'07)* (2007), pp. 177–178
172. N. Tintarev, M. Dennis, J. Masthoff, Adapting recommendation diversity to openness to experience: a study of human behaviour, in *User Modeling, Adaptation, and Personalization*, ed. by S. Carberry, S. Weibelzahl, A. Micarelli, G. Semeraro (Springer, Berlin, Heidelberg, 2012), pp. 190–202
173. N. Tintarev, J. Masthoff, Explaining recommendations: design and evaluation, in *Recommender Systems Handbook* (Springer, New York, 2015), pp. 353–382
174. W. Trost, T. Ethofer, M. Zentner, P. Vuilleumier, Mapping aesthetic musical emotions in the brain. Cerebral Cortex **22**(12), 2769–2783 (2012)
175. D. Turnbull, L. Barrington, D. Torres, G. Lanckriet, Semantic annotation and retrieval of music and sound effects. Trans. Audio Speech Lang. Process. **16**(2), 467–476 (2008)
176. D. Turnbull, L. Waldner, Local music event recommendation with long tail artists (2018). Preprint. arXiv:1809.02277
177. A. Vall, M. Dorfer, H. Eghbal-zadeh, M. Schedl, K. Burjorjee, G. Widmer, Feature-combination hybrid recommender systems for automated music playlist continuation. User Model. User Adapt. Interact. **29**(2), 527–572 (2019)
178. A. van den Oord, S. Dieleman, B. Schrauwen, Deep content-based music recommendation, in *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5–8, 2013*, Lake Tahoe, Nevada, United States, ed. by C.J.C. Burges, L. Bottou, Z. Ghahramani, K.Q. Weinberger (2013), pp. 2643–2651

179. S. Verma, J. Rubin, Fairness definitions explained, in *Proceedings of the International Workshop on Software Fairness, FairWare '18* (Association for Computing Machinery, New York, NY, 2018), pp. 1–7
180. G. Vigliensoni, I. Fujinaga, Automatic music recommendation systems: Do demographic, profiling, and contextual features improve their performance? in *Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR 2016*, New York City, August 7–11, 2016, ed. by M.I. Mandel, J. Devaney, D. Turnbull, G. Tzanetakis (2016), pp. 94–100
181. G. Vigliensoni, I. Fujinaga, The music listening histories dataset, in *Proceedings of the 18th International Society for Music Information Retrieval Conference*, Suzhou, People's Republic of China, 2017, pp. 96–102
182. D. Wang, S. Deng, X. Zhang, G. Xu, Learning to embed music and metadata for context-aware music recommendation. World Wide Web **21**(5), 1399–1423 (2018)
183. S. Wang, L. Hu, Y. Wang, L. Cao, Q.Z. Sheng, M.A. Orgun, Sequential recommender systems: Challenges, progress and prospects. CoRR abs/2001.04830 (2020)
184. M. Ward, J. Goodman, J. Irwin, The same old song: the power of familiarity in music choice. Market. Lett. **25**, 1–11 (2013)
185. D. Weigl, C. Guastavino, 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, October 2011
186. W. Wu, L. Chen, Y. Zhao, Personalizing recommendation diversity based on user personality. User Model. User-Adapt. Interact. **28**(3), 237–276 (2018)
187. S. Yao, B. Huang, Beyond parity: fairness objectives for collaborative filtering, in *Advances in Neural Information Processing Systems 30*, ed. by I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Curran Associates, Inc., Red Hook, 2017), pp. 2921–2930
188. H. Zamani, M. Schedl, P. Lamere, C. Chen, An analysis of approaches taken in the ACM recsys challenge 2018 for automatic music playlist continuation. ACM Trans. Intell. Syst. Technol. **10**(5), 57:1–57:021 (2019)
189. E. Zangerle, M. Pichl, W. Gassler, G. Specht, #nowplaying music dataset: extracting listening behavior from twitter, in *Proceedings of the First International Workshop on Internet-Scale Multimedia Management, WISMM '14* (Association for Computing Machinery, New York, NY, 2014), pp. 21–26
190. M. Zenter, D. Grandjean, K. Scherer, Emotions evoked by the sound of music: characterization, classification, and measurement. Emotion **8**, 494 (2008)
191. Y.C. Zhang, D.O. Séaghdha, D. Quercia, T. Jambor, Auralist: Introducing serendipity into music recommendation, in *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, WSDM '12*. (ACM,New York, NY, 2012), pp. 13–22
192. E. Zheleva, J. Guiver, E. Mendes Rodrigues, N. Milić-Frayling, Statistical models of music-listening sessions in social media. in *Int. Conf. on World Wide Web (WWW'10)* (2010), pp. 1019–1028
193. Y. Zheng, Context-aware mobile recommendation by A novel post-filtering approach, in *Proceedings of the Thirty-First International Florida Artificial Intelligence Research Society Conference, FLAIRS*, 2018, Melbourne, FL, May 21–23 2018, ed. by K. Brawner, V. Rus (AAAI Press, New york, 2018), pp. 482–485
194. Y. Zheng, R.D. Burke, B. Mobasher, Splitting approaches for context-aware recommendation: an empirical study, in *Symposium on Applied Computing, SAC 2014, Gyeongju, Republic of Korea - March 24–28, 2014*, ed. by Y. Cho, S.Y. Shin, S. Kim, C. Hung, J. Hong (ACM, New York, 2014), pp. 274–279
195. L. Zhu, Y. Chen, Session-based sequential skip prediction via recurrent neural networks. CoRR abs/1902.04743 (2019)
196. C. Ziegler, S.M. McNee, J.A. Konstan, G. Lausen, Improving recommendation lists through topic diversification, in *Proceedings of the 14th international conference on World Wide Web, WWW 2005*, Chiba, May 10–14, 2005, ed. by A. Ellis, T. Hagino (ACM, New York, 2005), pp. 22–32