Multimed Tools Appl (2017) 76:14375–14403
DOI 10.1007/s11042-016-3836-x

From manual to assisted playlist creation: a survey

Ricardo Dias[1] · Daniel Gonçalves[1] · Manuel J. Fonseca[2]

Received: 4 March 2016 / Revised: 25 July 2016 / Accepted: 2 August 2016 /
Published online: 18 August 2016

Abstract Nowadays, thanks to the popularization of music streaming services, we gained access to millions of songs to listen to. One of the methods employed by these services to support browsing and promote song discovery are playlists. Additionally, creating and sharing playlists over the Internet have become common practices. A playlist can be defined as a *"sequence of songs meant to be listened to as a group"*. Research on playlist creation has been done according to three perspectives: i) manual creation; ii) automatic generation and recommendation; and iii) assisted playlist creation. In this paper we review previous research on these three approaches, which we believe are complementary on the subject of playlist creation. We highlight the importance of combining insights from these three perspectives to better understand the current problems and methods, criteria and techniques, and how they complement each other. Furthermore, we identify promising research directions for the three different approaches of playlist creation.

Keywords Music playlists · Manual creation · Playlist generation · Assisted techniques · Survey

✉ Ricardo Dias
ricardo.dias@ist.utl.pt

Daniel Gonçalves
daniel.goncalves@inesc-id.pt

Manuel J. Fonseca
mjfonseca@ciencias.ulisboa.pt

[1] INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal

[2] LaSIGE, Faculdade de Ciências, Universidade de Lisboa, Lisbon, Portugal

1 Introduction

Digital music changed the way we listen to music. Over the last years, several streaming services like Spotify,[1] Rdio[2] or 8tracks[3] changed our conceptions and habits on music selection and listening. Thanks to these services, we gained access to millions of songs and music listening has become ubiquitous. Nowadays, we can listen to almost any song ever recorded, anywhere and at any time, using different devices. Like Fields reported in his work [37], *"the line between music that is yours and music you want to listen to fades"*. Indeed, this is one of the reasons why streaming services have become so popular.

However, the access to such vast and diverse music collections did not come problem free. Selecting and filtering songs has become a strenuous task, as people feel doubtful about which songs to listen from such huge collections [79]. As a way to solve this problem, streaming services make use of playlists to promote song discovery, and as an entry point to help users browsing the huge amount of available songs. Moreover, these services allow playlist creators to collaborate and share their playlists with friends and followers. Playlists are used not only to aggregate songs about a specific artist, genre, purpose or theme, but also as a way to leverage the burden of having to browse these music collections through the different perspectives of their content and to promote the social status of the playlist creators.

During the last decade most efforts have been directed at the creation of algorithms for the automatic generation of playlists, without human intervention. The pioneer work by Pauws and Eggen [75], or the recent techniques based on different users' facets and listening habits like, for instance, the temporal context [47] or daily activities [34] are some examples. Despite the use of different data, their goal remain the same: to capture and encode the listener tastes and habits to recommend songs and generate playlists. Bonnin and Jannach describe that comprehensively in their state-of-the-art survey on automated playlist creation [14]. Nevertheless, researchers have turned their focus to the role of human intervention in playlist creation, striving to understand the creators behaviors, needs, techniques and interaction [23, 52, 78]. Findings from these studies evidence that user feedback should be included in the techniques to improve them, and also to engage users [93], because they enjoy creating playlists [52, 60], and as listeners they trust and give preference to handmade playlists created by them or others, over automatic playlists or recommendations [6, 9, 59]. This preference is mostly related to emotions, the feel that someone spent time preparing the playlist or performing the recommendation, or that they selected the songs for some special purpose or occasion. One successful case that evidences this, is the *8tracks* website, that currently has more than eight million active users monthly and contains thousands of handmade playlists, with songs carefully picked to convey the creators message. The subscribers of this service enjoy listening to playlists created by others and also to create new playlists. This is a tendency in music consumption as researchers and industry realize the importance of the personal touch in playlist creation.

In this paper we survey the topic of playlist creation according to three perspectives: **manual** (individual song picking), **automatic** (no human intervention) and through the use of **assisted** techniques (guided and visual creation). We believe these are complementary

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

[2] http://www.rdio.com

[3] http://www.8tracks.com

approaches for playlist creation, from which we can get insights to understand the current problems, methods, criteria and techniques employed.

Manual creation is the most basic and old approach for playlist creation, as it is the simplest (but not the easiest) way of creating playlists through individual song picking and selection. Our goal on reviewing work on this topic is to identify criteria, methods and habits on playlist creation regarding time, effort, goals and purposes.

Automatic approaches can generate playlists almost without human intervention, leveraging the effort required by creators in the process. Both Sneha et al. [83] and Bonnin et al [14] presented updated surveys on automated playlist, focusing on the algorithms and mechanisms applied for playlist generation. Therefore, here we summarize their works and correlate them with manual creation techniques. Whenever required we refer to these papers for a more complete research on automatic generation.

Finally, assisted techniques engage users in browsing and playlist creation tasks by assisting them throughout the process using visualization techniques [93]. In this topic we studied interactive visualization techniques to withdraw insights on the use of such techniques to ease playlist creation.

These three approaches cover different facets of playlist creation and provide a broader understanding on how to create playlists, the current trends and their limitations. To the best of our knowledge, no broad analysis covering different perspectives, as we present here, has been made. Most surveys about playlist creation focused on a single perspective (automatic generation or recommendation) [14].

In the remainder of this paper we start by describing in Section 2 the background for playlist creation, defining the concept of playlist, the types of playlists and desirable properties for them. In Section 3 we review the state-of-the-art work in playlist creation, focusing on manual, automatic and assisted creation. Section 4 summarizes and discusses the advantages and limitations of the three approaches. Finally, in Section 5 we conclude the paper and present possible research paths to explore.

2 Background

In this section we describe the background for understanding playlist creation techniques. To this end, we provide an incursion throughout history to evidence the origin of the term *Playlist*, and end up with a clear definition for the concept. Furthermore, we discuss the different types of playlists and their desirable properties as found in the literature.

2.1 History of the playlist

Figure 1 depicts the key moments in the history of the Playlist since the times before recorded music to the current digital era.

The term *playlist* was first used around the beginning of the 20th century during radio dissemination, to describe an *unordered set of songs* [16]. However, its origin goes back to the period of non-recorded music, when related concepts begun to take shape. By this time, around 1850's, the concepts of **mixing** and **coherence** within concert programs started to gain relevance [58]. Instead of assembling musical pieces to maximize coverage of taste, they were selected to convey and express the intentions of the program director, leaning towards his **personality** and **preferences**.

During the late 19th and early 20th centuries, **radio broadcasting**, and **audio recording and playback** inventions led the next step towards the present day. These two technological

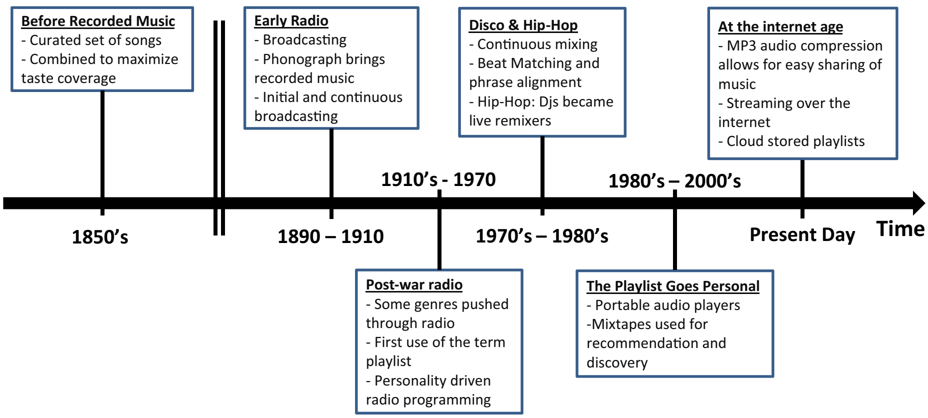


Fig. 1 The playlist history

innovations made music accessible to a larger audience without the physical presence of the performers. As radio began to be more disseminate, some genres like Rock & Roll and R'n'B emerged and become more pushed [90]. Once again, like what happened while assembling musical pieces to concert programs, much of the programming in radio was driven by **hosts' personality**, their tastes and preferences. Nonetheless, by that time, song selection was also heavily influenced by **external constrains**, like for instance, the sales.

Back in the mid 1970's, Disc Jokeys (DJs) introduced the concepts of **continuous mixing** and the *"elimination of space between songs played back in sequence"*, which have remained used till the present day. Later on, *continuous mixing* was pushed further with the rise of the hip-hop culture, where DJs become *live remixers* and the turntable their instrument [2]. By the same time, portable audio devices emerged (specially the Walkman), which drove to the popularity of *cassette tapes* [36]. The usage of tapes became wide spread, allowing the **combination** and **reordering** of different songs into personal mixtapes [17]. Mixtapes were therefore used as a mean for recommendation and discovery, as everyone could record and distribute them socially. People were able to record mixtapes in the order of their preference.

Along with the change from the analogical (or physical) to digital age, the playlist took a further step to its current understanding. With the formation of the World Wide Web and the psychoacoustic audio compression (MPEG layer 3), **sharing mixtapes** *without the physical constrains* became possible [39]. Moreover, besides the **personal exchange** of songs and mixtapes, **radio streaming** became a reality (broadcasting radio over the WWW), although they have remained similar to the traditional terrestrial broadcast for a while. Nonetheless, online broadcasting made it possible for a station to reach more people in the world, while at the same reducing the cost for a new station to emerge.

Nowadays, the current trend is towards common **web-based storage of playlists**, where streaming services either provide listenable content or links to buy them, and promote **sharing** and **discovery** behaviors. Some examples include Spotify, Rdio, 8tracks or more recently the Apple Music.[4]

In short, despite all the technological advances during the last century, the definition of *playlist* is still under discussion till present day. The analysis of the history of the playlist

[4] https://www.apple.com/music/

depicted here shows us the emergence of several concepts that are still nowadays related with playlists, such as: i) mixing and coherence of songs in playlists; ii) ordering and combination to maximize satisfaction (or profit in some cases); iii) influence of individual preferences / tastes for song selection; iv) the need for smooth transitions to maintain a continuous flow.

In the following section we further discuss the definition of playlist.

2.2 Playlist definition

The concept of playlist has been changing and evolving since its origin until current days, and although there have been indecisions about its definition throughout the literature (and their nomenclature), it is noteworthy the definition presented both by Fields [37, 38] and Bonnin et al. [14], as it stands out as the most commonly accepted one. According to these authors, a playlist can be described as an ***"ordered sequence of songs meant to be listened to as a group"***.

This definition highlights three key aspects of a playlist: 1) it consists in a **set of songs**, 2) the songs have an **explicit order**, 3) they are intended to be **listened as a whole**. Notice that although the songs have an explicit order, its importance has decayed over time, because nowadays most listeners play them in random or shuffle modes [62]. On one side, this shuffling behavior has been influenced by the playback policy of streaming services, such as those imposed by 8tracks that shuffles a playlist after the first time we listen to it. On the other side, the shift to the digital era of the collaborative and sharing platforms, gave creators the freedom to easily change the order of the songs as they please, and for listeners to choose how they would reproduce the playlist.

Playlists have also inherited some characteristics from mixtapes (or mixes). In [30], the authors distinguish between *mix tape and playlist*. For these authors a mix tape is usually considered as a set of songs with some **defined length**, a strong **defined theme**, where **order is important**, and is often **for sharing**, while the playlist is more **personal**, varies in *length* and is **less structured**. This differences are linked back to the time of physical mixtapes, when songs were recorded in *CDs* or *Cassette Tapes*, that imposed a maximum number of songs and a fixed order. Nevertheless, in our increasingly digital world these two concepts share most of the pinpointed characteristics. Thereby, we complement the definition presented above, by providing a more complete definition of playlist:

> *"A playlist is an arbitrary sequence of songs meant to be listened to as a group and that fit a certain theme or purpose either for individual reproduction or sharing."*

This definition is brief enough to remain simple, but still flexible to accommodate the characteristics of different types of playlists as described in the following section.

2.3 Types of playlists

A recorded mixtape, a pre-recorded DJ set/mix, a live DJ set, a radio show log, or an album are just a few examples of playlists that fit our definition. In fact, there have been attempts in the literature to categorize these different types of playlists. Two complementary categorizations have been performed by Fields [37, 38] and Bonnin et al. [14].

Fields classifies playlists according to the relationship between the **creator** of the playlist, the producer, and the **listener** of the playlist, the consumer. Four entities are considered in his approach (both producers and consumers): 1) the expert, someone who lives or makes money for creating playlists; 2) the listener, the person that listens to the playlists; 3) the peer, an amateur that creates playlists; 4) the self, to identify the same entity in specific contexts. By using this taxonomy different playlists can be classified by simply adapting the roles of the producer and the listener.

Bonnin et al. categorize playlists into six types: 1) Broadcast radio playlists, 2) Personalized radio playlists, 3) Amateur playlists, 4) Club playlists, 5) Album tracklists and 6) Compilations tracklists. This categorization implicitly relies on the relationship between the creator of the playlist and the intended target or goal. For instance, **broadcast radio playlists** are created by experts in radio stations for the listeners of that radio, and *amateur playlists* are created by regular people either for other peers, or even for themselves.

Despite minor differences, by carefully analyzing these two categorizations, we unveiled associations between them. In Table 1 we correlate the two perspectives, providing a description of the playlists types we consider throughout this research.

2.4 Desirable properties for playlists

Understanding and unveiling the properties of a *good playlist* is important to improve current methods and techniques for creating them. For example, the preference or familiarity with some songs, the structure and diversity of the playlist, or the songs freshness and coolness, are factors that influence the perceived quality of a playlist [12, 37]. Back in 1997, de Mooij [31] conducted an evaluation with 14 participants to ascertain which of eight factors had the biggest effect on a playlist. The results of this evaluation revealed that*song selection* was rated by users as the most important factor. However, the work left open the question of what makes the songs right for the playlist, in particular, the idea of musical taste for the long-term, slow-to-change preferential commitment to a genre that can have an impact.

Elements of the *order* in which the songs are arranged in a playlist also have an impact on perceiving the quality of a playlist. *Song transitions*, the overall *structure* of a playlist and the occurrence of *serendipity* are some properties that evidence the importance of order [31].

Other elements can have a further effect on preference, especially a listener's mood and the context (i.e. activities, time, location, etc.). One particularly important factor is how *familiar* a listener is with a song. It has been shown that people will often prefer listening to familiar songs that they like less than non-familiar songs [37].

In a recent work, Jannach et al. [50] analyzed the characteristics of thousands of playlists created and shared online. They relied on the musical and meta-data features to understand desirable characteristics of playlists. The most desirable properties discovered by the authors were: **Popularity, Freshness, Homogeneity and diversity, Musical Features, Transition and Coherence**. Notice that most of these findings are the same that were found

Table 1 Characterization of the different playlist types based on the relationship between the creator and the listener of the playlist

Playlist type	Entity relationship	Description
Broadcast radio playlists	Expert to listener	Made by DJs in radio stations
Personalized radio playlists	Peer to peer	Generated by Web Music Services
Amateur playlists	Peer to peer; Listener to self	Made by nonprofessional music enthusiasts
Club playlists	Expert to listener	Playlists made by DJs in clubs
Album tracklists and compilations tracklists	Expert to listener	Made by Artists or Labels

in previous studies [31] and throughout the history of the playlist as described in Section 2.1. Thereby, we provide a brief description of their findings and some possible applications of these properties:

- **Popularity**: Whenever we talk about popular songs in a playlist, they should come first in playlists. They should not be *"too obvious"* nonetheless, but instead provide a *"great opening"* for the rest of the playlist.
- **Freshness**: Playlists should be homogeneous regarding *freshness* (the release date). Results evidence that listeners prefer recent songs and playlists are often quite uniform with regard to freshness. However, other studies reveal that this preference is related to other constraints, such as age[5],[6].
- **Homogeneity and diversity**: Creators tend to keep the playlist homogeneous regarding both artist and genre. However, balancing homogeneity with song diversity is mandatory to maximize the listeners' satisfaction.
- **Musical Features**: Creators value musical features differently. Findings from these studies identified *Energy* and *Hottness* as the most relevant ones. Danceability, Loudness and Tempo were not considered very important.
- **Transition and Coherence**: Much like maintaining a playlist homogeneous, maintaining smooth transitions and coherence are two other key properties of good playlists. Nonetheless, the findings of these studies [14] suggest that usually the second halves of the playlists have a lower coherence than the first halves, which might denote that the songs of the second halves seem to be slightly less carefully selected than those of the first halves.

In summary, although assembling a playlist can be seen almost as an art, good playlists share some intrinsic properties, like for instance, a smooth flow between songs, or a strong coherence and homogeneity. Nonetheless, more research is required to fully understand what makes a good playlist. Indeed, it would be interesting to compare the importance of those properties for both the creators and listeners of the playlists.

3 Playlist creation techniques

Different techniques have been applied and developed to support playlist creation. In this section we analyze the state-of-the-art research in playlist creation according to three perspectives: 1) manual (individual song picking), 2) automatic (no human intervention) and through the use of 3) assisted techniques (guided and visual creation). These three complementary approaches cover different facets of the playlist creation process and provide a broader understanding on how to create playlists.

3.1 Manual playlist creation

Ever since the days of the radio, manual playlist creation has been the way of creating playlists over the years [37]. Generally, it consists n purpose or goal, to convey a message

[5] http://musicmachinery.com/2014/02/13/age-specific-listening/

[6] http://skynetandebert.com/2015/04/22/music-was-better-back-then-when-do-we-stop-keeping-up-with-popular-music/

or to express feelings [30]. In the following subsections we look at the details about this method of playlist creation.

3.1.1 Styles

Overall, creators start by thinking about a few particular songs to add to a playlist and proceed from that by browsing the music collection, either filtering by album, artist, genre, or even using similarity between the artists, as it happens nowadays. Then, when they have all the songs they want in the playlist, they can reorder them to get a better flow and transition between songs [30]. Most music players and streaming services support this approach, by letting users browse their collections using traditional list-based visualizations.

Other playlist creation styles have been identified by Lehtiniemi et al. [60] during an experimentation with a playlist generation approach. Table 2 summarizes these styles. Notice that while some require more effort and interaction from the users (like Style 2 playlists), others evidence less attention to the playlist creation process (like Style 3 playlists). This categorization suggests different playlisting behaviors and profiles. So far, to the best of our knowledge, no research have been performed to analyze the different profiles of playlist creators.

3.1.2 How people create playlists

Although the description of the manual playlist creation process seems to characterize manual playlist creation as being simple and straightforward, there are several hitches hidden in it, that are essential to understand the creation process, namely: i) what are the purposes for creating a playlist (and how these influence the choice of the songs); ii) how do users initiate browsing tasks and what criteria do they use to select songs; iii) what are the roles of the properties that make a good playlist (such as, order and song position, flow and transitions) in manual creation.

Cunningham et al. [30] presented an analysis about playlist creation (and mixes) by collecting data on how people manually create and organize playlists. The findings from this study revealed that people create playlists for several reasons, namely: i) to serve as the background for other activities (such as traveling, studying or exercising at the gym); ii) to convey or express an emotion; iii) to be used in an event, like for instance, a party or a wedding.

These authors also find interesting results regarding the order of the songs and the length of the playlist. Their findings unveil that though the order of the songs is *"usually*

Table 2 Different styles of manual playlist creation [60]

Description	
Style 1	User chooses full albums for the playlist with the original song order and wants to hear the whole album as an entity
Style 2	User chooses individual songs from music catalog (with individual songs from different artists randomly ordered) for playlists
Style 3	User chooses all songs from the music catalog to the playlist and plays them randomly
Style 4	User chooses individual songs from music collection (music collection contains full albums) to the playlist

significant", there are no clear rules for sorting the songs in a playlist. Despite that, participants presented some rough descriptions for it: i) there cannot be more than two songs from the same artist or genre in sequence, unless there is "a special link" between the songs; ii) consecutive songs should have complementary styles or sounds, to avoid the *"clash of one song against another"* during the mix; iii) the first song on the playlist should be good, however, not the best one of the list; iv) the final song should be carefully selected, as it might leave a *"pleasant memory of the list"*. This evidences different rules for playlist creation and that they are not indeed very strict and clear, indicating that more research is required to get a deeper comprehension. Regarding the average length of the playlists, their findings confirm what we have describe in Section 2.2, about the differences between old mixtapes and playlists. Though in the past, the length of a playlist used to be linked to the medium used for recording it, like tapes or CDs with few songs (up to 20 songs on average), in current digital environments, either streaming services or personal collections, the length of playlists can be completely arbitrary, depending on its purpose.

Stumpf and Muscroft [84] conducted a study to analyze how users create playlists for different music listening contexts: 1) a Large Party, 2) a Small Gathering and 3) a Private Travel. The authors investigated what features users apply to characterize music and how the context influence the attention put on these features. To achieve their goals, they identified shared terms (or features) used by users (both listeners and creators) to describe music, and how the listening context influenced the attention given by users to that features.

According to these authors, most of the participants do not share exactly the same vocabulary to describe the songs. Once again, this is indeed a fact that makes the playlist creation process harder. Despite that, they were able to distinguish between two categories of features from the descriptions given by the participants: 1) *intrinsic song characteristics*, which are based on *"what the users know about the songs alone"*; ii) *context-related characteristics*, which are related to the context where the songs are about to be listened, assuming that listening to music is not the primary task. Intrinsic song characteristics are mainly about audio **content-based features**, namely: Tempo, Rhythmic Quality, Composition, Texture (about the different voices in songs, for instance, male vs. female) and Volume. Regarding **context-related** characteristics, the authors highlight the prominence, social catalyst and audience type features. Their findings revealed that: i) Tempo and Genre may be relatively context-independent; ii) Mood is an important feature for the Private Travel use case; iii) Popularity and Age are more important for use cases involving other listeners (Large Party and Small Gathering); iv) Rhythmic Quality was frequently used to describe the Large Party use case.

3.1.3 Advantages and drawbacks

Manual creation is not problem free. Indeed, it is time consuming, with creators taking a lot of time selecting and discovering the best songs to include in a playlist [14]. This is even worse nowadays, because people are using more and more streaming services, and therefore, having access to larger music collections. In fact, the choice from endless songs [79] is a difficult problem to solve when using manual methods only. This is clearly an evidence that complementing manual creation with automatic suggestions, can be a possible improvement for playlist creation solutions. Moreover, these techniques can ease the burden of song selection carried by creators, while enabling their playlists to promote song discovery.

In short, manual song picking might be the simplest approach for assembling a playlist, but it is a complex subject with several problems that require attention from researchers. To solve some of them, like for instance, promote song discovery (and serendipity), researchers have turned their attention to automatic solutions, which are less expensive, less time-consuming and can be used to adapt to different listeners' tastes and behaviors. The following section describes the state of the art research in automatic music playlist generation.

3.2 Automatic playlist generation

Automatic or automated solutions for playlist generation have received the most attention and effort from researchers during the last decade. These mechanisms can adapt to different users' tastes and listening habits, by capturing preferences about each individual behaviors and therefore, easing the burdens of manual creation. Indeed, it is the absence of human intervention along with the development of more powerful machines that made this type of techniques so popular, where the machines become the creators and us the listeners.

Research on these techniques has been essentially performed on the algorithms that encode preferences and tastes and perform the suggestion of sets of songs. In the following sections we go further in the analysis of this type of approaches for playlist creation.

3.2.1 The playlist generation problem

When we argue about automatic or automated playlisting techniques it is relevant to use the term **playlist generation** instead of **playlist creation**. In these techniques, because almost no human interaction is used (or almost any interaction), it is named *generation*, as the focus are the algorithms that perform the creation.

"The playlist generation problem typically consists in creating such a list given either some seed information or semantic description" [13]. Formally, one can describe the problem as how to create a sequence of tracks fulfilling a set of target characteristics in the best possible way, using a pool of tracks and a background knowledge database [14]. The seed information can be an ordered set of tracks corresponding to the creators' listening history [33], and the target characteristics a description of the purpose of the playlist constrained by the desirable properties of a good playlist (see Section 2.4). On popular platforms such as *Last.fm*, for example, sequences of songs are automatically generated starting from some seed song or artist. Other sites such as *8tracks* allow users to share their manually created playlists with others. In either case, the playlists usually contain at least some items which are novel for the listener [12], or suitable for a specific purpose.

There are a number of aspects that make the playlist creation task challenging, namely: 1) the large number of available tracks (large collections); 2) the issues on acquiring information about the tracks; 3) the taste and preferences of the listeners; 4) the context for which the playlist is going to be created. In the following we discuss the strategies used in state-of-the-art playlist generation techniques, to address these challenges.

3.2.2 Playlist generation techniques

A considerable number of techniques to automatically generate playlists have been proposed during the last years. Bonnin et al. [14] and Sneha et al. [83] both presented surveys about automatic solutions for playlist generation. These authors give details about the generation process and algorithms applied, the challenges that make the topic interesting, a categorization and description of the different solutions to address playlist generation, highlighting the main advantages and drawbacks. Since these surveys make a thorough and up-to-date characterization of this subject, here we summarize their insights, referring to their work whenever we require a closer analysis.

These authors classify playlist generation algorithms into seven categories (Fig. 2): 1) Similarity-based Algorithms; 2) Collaborative Filtering; 3) Frequent Pattern Mining; 4) Statistical Models; 5) Case-Based Reasoning; 6) Discrete Optimization and 7) Hybrid Techniques. Table 3 summarizes the advantages and drawbacks of each solution, while in the following paragraphs we provide a brief description of each category.

Similarity-based algorithms Similarity-based approaches rely on the closeness or similarity between songs, using a distance function (like for instance, the Euclidean distance [53]), to generate playlists. Different data can be used as input for the distance function, like for instance, features extracted from the audio signal or metadata, like the artist or genre [82]. Some strategies for selecting the most suitable songs for a playlist, include for instance: i) selecting songs based on their distance to a seed song; ii) according to the similarity with the previously selected tracks; iii) selecting similar songs to the favorite ones of a particular user.

Collaborative filtering Collaborative-filtering (CF) methods have been the predominant approaches in the field of Recommender Systems, and they typically work by learning past user listening behavior, and recommending songs to a user based on ratings of those songs by other users with similar tastes [11, 27, 56]. Although most of the collaborative filtering techniques were not specifically designed for playlist generation, they can be applied for that purpose by changing the data used in the *user-item rating matrix*.

Frequent pattern mining Frequent pattern mining approaches for playlist generation are based on a principle of neighborhood, considering both *local* and *global* patterns in the data, which can be grouped either using **association rules** and **sequential patterns**. Typically, *association* rules are applied when the order of the elements in the pattern is not relevant, while sequential patterns are used when the order of elements is relevant. Choosing between these two types of rules depends on the data characteristics [26, 28]. To automatically perform playlist generation, frequent patterns can be extracted from manually created playlists.

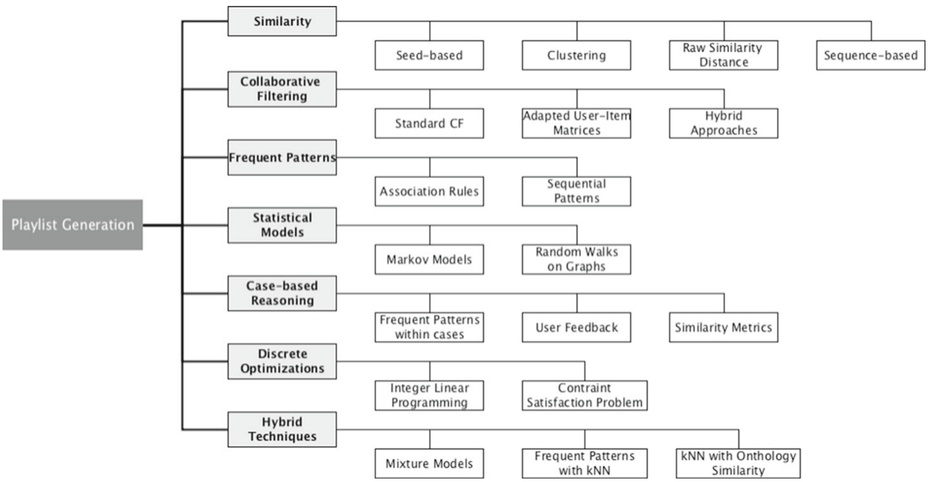


Fig. 2 Overview of the most common techniques for playlist generation. For details about the algorithms and their parametrization, please refer to Bonnin et al. [14]

Table 3 Advantages and Drawbacks of the different techniques for automatic playlist generation

Automatic creation techniques	Advantages	Drawbacks
Similarity-based	Scale to large music collections; creates homogenous playlist	Promotes little diversity and song discovery
Collaborative filtering	Extensive research on collaborative filtering methods; Adapts to past preferences	Requires many data to perform accurately
Frequent pattern mining	Generated playlists can implicitly reproduce the observed characteristics of manually defined playlist	Quality of the playlists generated depends on the number and quality of the playlists used for pattern mining
Statistical method	Plenty of algorithms for optimizing the playlist generation	Learning process of these algorithms can be time consuming
case-based reasoning	Low computational complexity when a limited number of cases is used	Do not scale well for repositories with many cases
Discrete Optimization	Generation can satisfy most of the target characteristics, when background knowledge is accurate	Most solutions are computational complex and expensive; scalability issues
Hybrid techniques	Combination of different techniques to overcome individual limitations; Can adapt to different contexts	Can be more expensine and time consuming than a sub-optimal solution

When some songs appear together in these playlists, one can assume that they share some characteristics and that they fulfill the intended purpose of the creator.

Statistical models Another approach for automatic playlist generation is through the use of statistical models. Markov models are the simplest algorithms used in the approaches. For these models, one assume that each song is only dependent on the previous one, and can estimate the transition probabilities applying different strategies, like for instance: i) the co-occurrence of songs in the playlists, ii) content-based similarity or metadata.

Case-based reasoning Case-based reasoning solutions explore information about past problem settings (named cases) to solve new problems [1]. Furthermore, these approaches work first by storing a set of representative cases along with their solutions in the *case repository*, and then, given a new problem instance, search for similar cases that might help solve the current problem.

Two relevant case-based reasoning solutions can be found in the literature [4, 5]. In the first work the researchers presented an approach to find the playlists considered to be the most *useful*, instead of finding the most *similar ones*, based on a given seed track. The same authors proposed another technique for a broadcasting radio, where user preferences are mapped in cases, to adapt the selected tracks for a target set of listeners.

Discrete optimization Another approach for automatic playlist generation is to see it as a discrete optimization problem. The goal of these approaches is to create a playlist with songs that satisfy a set of defined constraints. Different constraints can be used, like for instance, the transitions between songs, or a given start / end song.

Some examples of playlist generation techniques using discrete optimization include the work developed by Aucouturier and Pachet [3], where the authors propose creating playlists by iteratively enhancing a randomly chosen playlist through a local search procedure [49, 76].

Hybrid techniques Hybrid techniques have been used to overcome individual drawbacks, by combining the advantages of the different techniques employed. They have been extensively researched over the last years in the field of Recommender Systems [23] and specifically for playlist generation.

Hybridization methods can be classified into seven categories as depicted in Table 4, namely: Weighted, Switching, Mixed, Feature Combination, Cascade, Feature Augmentation and Meta-level. Though not initially designed for playlists generation, some examples of hybrid methods can be found in the literature, like for instance, the work developed by McFee and Lackriet [67], where the authors used a weighted hybridization approach, combining algorithms based on song similarity and song familiarity. Hariri et al. [45] also used a weighted strategy that combined a frequent pattern mining algorithm with a k-Nearest Neighbors (kNN) approach.

3.2.3 Advantages and drawbacks

Automatic solutions can adapt to users tastes, leverage the effort of song selection from the millions available, and promote song discovery. However, they are not problem free. These techniques are focused on the algorithms used and they lack control and transparency during the creation process. Most of these solutions are independent of user interaction and act like black boxes, hiding the rationale of the decisions behind the suggestions performed. These two factors, control and transparency, have been studied over the last couple of years, with results revealing that they are desired features for recommender systems [54, 87]. Studies have confirmed that solutions which induce users with the sense of control and transparency, increase the trustiness that they have in them, even when they do not perform so well in typical system-based evaluations (like for instance, in the task of predicting user ratings for songs) [54].

Assisted approaches that apply visualization techniques to support a manual playlist creation and control the over automatic algorithms stand out as a possible solution for creating

Table 4 Classification of hybridization methods proposed by burke [18]

Hybridization method	Description
Weighted	The scores (or votes) of several recommendation techniques are combined together to produce a single recommendation.
Switching	The system switches between recommendation techniques depending on the current situation.
Mixed	Recommendations from several different recommenders are presented at the same time.
Feature combination	Features from different recommendation data sources are thrown together into a single recommendation algorithm.
Cascade	One recommender refines the recommendations given by another.
Feature augmentation	Output from one technique is used as an input feature to another.
Meta-level	The model learned by one recommender is used as input to another.

playlists, because they give control to users and make collections and algorithms transparent [87]. In the following section, we review research on playlist creation using interactive assisted approaches, highlighting the use of different techniques and the application of both manual playlist creation and automatic algorithms.

3.3 Assisted playlist creation

On one side, users enjoy creating playlists manually and give preference to handcrafted playlists [52], but this is time consuming and impracticable for large collections. On the other side, automatic algorithms can model users' tastes and preferences without their intervention, and perform suggestions based on past listening habits, leveraging the burden and time spent for song selection. Nonetheless, the latter solutions remove control over the creation process from users and lack transparency. Although both approaches for playlist generation have advantages and drawbacks, we can benefit from each other, by developing techniques that can set a bond between manual and automatic methods. Interactive techniques for an assisted (or semi-automatic) playlist creation seems the natural path to follow [40].

During the last decade, researchers have developed and studied different visualization techniques for browsing and representing music collections, based on the structure of the music library, data from the songs (both content features or metadata [22]), external data from the web or the habits and behaviors of users. They support playlist creation and provide different perspectives of the underlying music collection through browsing techniques. Furthermore, interactive visualization techniques present various benefits for playlist creation: i) they are interactive; ii) provide control during the creation process; iii) make the creation task more transparent and iv) engage users. In the following we discuss these properties according to playlist creation.

3.3.1 Interactivity

"The central feature of recent computer visualization systems is interactivity (...) and you can add interaction to anything" [35].

Static visualization techniques can provide representations of music collections, suggested songs, etc. to support either the mapping of the underlying data or to allow users to get insights from them [19]. They make it visually appealing and possibly unveil new knowledge from the data [8, 33].

However, it is interactivity (responsiveness) that allows users to engage in playlist creation by interacting with the visualizations, manipulating and adapting filters, changing parameters, etc. As Kosara et al. [57] reported, *"The user is able to understand the information better, if the representation is not simply static, but rather he or she can interact with it"*, making interactivity essential for playlist creation.

3.3.2 Control

Control, or the sense of control, has a major role in playlist creation. Creators like to have control over their actions [81] and they trust in solutions that provide them with a sense of control [52].

Interactive visualization techniques engage people in browsing tasks by putting them in control [65, 93]. These techniques expose mechanisms for selecting and filtering songs, positioning and reordering songs, etc. They support control for manual creation, but also for managing and interacting with automatic approaches, like for instance, adjusting their parameters, or providing feedback [15].

Though control is a good feature of these techniques, it is important to bear in mind that different levels of control for playlist creation should be supported, each depending on the purpose of the playlist [40].

3.3.3 Transparency

Not only visualization techniques allow users to interactively control the playlist creation process, but they also make it transparent by providing visual representations of the songs, the music collection or even recommendations / suggestions [73].

Previous research in recommender systems have shown that visualization techniques improve transparency in several aspects [87], like for instance, it can visually combine multiple sources of recommendations or provide explanations about those recommendations. On music collections, it allows users to get insights about the content of the collection, providing a visual representation of it [32, 85]. Moreover, previous studies provide evidence that solutions which induce users with the sense of control and transparency, increase the trustiness that users have on them [54].

3.3.4 Engagement

"People often have no well-defined needs and rarely look for one specific item. Furthermore, most of their time is spent in browsing-alike activities and they may identify the match when they see it" [24]. Furthermore, as described by the authors, the pleasure of seeking and discovering new songs is usually the driver behind this browsing behavior, and not an information need [93].

When selecting songs for a playlist, exploratory behavior is a common practice as we discussed in Section 3.1, and highly interactive solutions that give users a sense of control, engage them in exploratory tasks [65].

3.3.5 Techniques

During the last decade, several approaches for supporting playlist creation have been developed using visualization techniques. In the following subsections we describe research for the different visualization techniques, focusing on how they are created, on how the features are used for it, and how users are assisted to create playlists with them. Whenever appropriate we relate them with the manual and automatic creation approaches. In Table 5 we summarize the most prominent techniques developed, by performing a categorization according to the type of the main visualization technique employed and the features they use.

Maps Map visualizations are useful techniques to represent large sets of data, by providing an overview of them. Several map visualizations can be extensively found in the literature and they have been applied for representing music collections and supporting playlist creation at least during the last decade. Most used techniques include *Self Organizing Maps (SOM)* [55], *Treemaps* [80], and other non-typical techniques [25, 46, 61]. Though some map visualizations, like for instance Treemaps, can be the visual representation of a node-and-link dataset (a graph), we decided to separate the techniques in terms of the visualizations generated and not the data used.

Table 5 Assisted techniques for playlist creation grouped by the type of visualization used

Type	Work	Features used	Playlist creation methods
Map	Islands of music [71]	Acoustic features	Seed-based; Path drawing
	Torrens et al. [85]	Metadata	Manual selection
	MusicMiner [69]	Acoustic features	Manual selection
	PlaySOM [70]	Acoustic features	Manual selection; Path drawing
	3D Islands of music [53]	Acoustic features	Seed-based; Path drawing
	Globe of music [61]	Acoustic features	Seed-based
	MusicBox [64]	Acoustic Features	Path drawing; Similarity-based
	Mobile music explorer [42]	Listening History	Similarity-based; Path drawing
	MusicSim [25]	Acoustic features	Similarity-based
	MuVis [32]	Acoustic features	Manual Selection; Similarity-based
	Geoshuffle [68]	User paths; Acoustic features	Similarity-based
	Musicovery	Unknown	Similarity-based
Graphs	Crampes [28]	Song sequence	Statistical models
	RAMA [43]	Metadata	Manual selection
	Carreira [21]	Acoustic features; Lyrics	Manual selection; Similarity-based
	Smarter playlists	Metadata	Manual parametrization; Similarity-based
Dots	Musicream [41]	Acoustic features	Manual selection; Similarity-based
	Vignoli [89]	Rhythm; Mood; Genre; Year	Path Drawing
	MOODetector [20]	Acoustic features	Similarity-based; Path drawing
Radar	AudioRadar [48]	Speed; Rhythm; Tone; Mood	Manual Selection

Self Organizing Maps [55] have been one of the most used map visualizations. Islands of Music (IoM) [71] is a content-based visualization technique for music collections. This technique displays songs in a virtual 2D map with islands and seas, using the SOM to determine the placement of songs in the map, and the Smoothed Data Histogram (SDH) [72] for calculating the topology of the map. In this visualization water represents the absence of songs, and the islands a concentration of songs. IoM uses a similarity metric between songs determined based on acoustic features from songs. A similar approach was developed by Knees [53], which uses a three-dimensional visualization of the islands and seas metaphor presented by Pampalk. Again, similarity between songs was determined using content features (the *rhythm patterns*). To facilitate the recognition and browsing mechanisms, the authors enriched the map with web-based keywords and artwork from the albums. Although these techniques were not specifically designed to support playlist creation, they allow the creation of playlists using both similarity-based approaches or through path drawing on the maps. Other SOM-based techniques include the PlaySOM and the PocketSOMPlayer [70], and the Globe of Music [61]. PlaySOM and PocketSOMPlayer are two novel interfaces to browse music collections by navigating maps of clustered songs based on content features. The former was primarily designed to allow interaction with large-screen devices, whereas the latter was implemented for mobile devices. Two modes for playlist generation are supported in these approaches: 1) rectangular selection to select entire clusters of music; 2) path drawing, to create playlists with smoother transitions between genres. Both playlist generation methods are based on song similarity. Globe of Music [61] is a solution for browsing and visualizing music collections through the use of a 3D globe. Each vertex in the structure represents a song in the collection (using its artwork), and the songs are placed according to their similarity using acoustic features. Though playlist generation was not the authors' main goal, one can rely on seed-based methods (similarity) to generate a playlist.

MusicMiner is another SOM-based approach for representing a music collection [69]. Playlist generation can be performed by manually selecting songs from the map. Nonetheless, even though content-based features are used, automatic playlist generation using this information was left out from the approach.

MusicBox [64] is an interactive music collection browser that represents the music collection in a two-dimensional space. Songs are mapped to the visualization by applying PCA (Principal Components Analysis) to contextual (genre, artist) and content-based features (rhythm, timbre, etc.). Similarly to other works, this approach allows users to generate playlists by creating a path between songs represented in the map.

Maps have also been used in mobile contexts. For example, Goussevskaia et al. [42] proposed a map visualization for a mobile solution that supports browsing and playlist creation by letting users draw a path over the map. The visualization is generated by using song similarity derived from past listening preferences, extracted from public sources. GeoShuffle is another content-based music browsing and exploration for mobile devices [68]. The authors propose a map visualization called *selforganizing tag clouds* (based on a SOM), *"a 2D tag cloud representation of a self-organizing map calculated using audio features"*. Moreover, GeoShuffle takes advantage of the mobile context (time and location information) provided by smartphones. Playlists are generated based on the location of the user, path and past listening preferences.

In MusicSim, Chen and Butz [25] presented a prototype for browsing and organizing large music collections that integrates audio analysis with user feedback into a highly interactive user interface. Along with other approaches, the 2D visualization clusters similar songs according to acoustic features. Additionally, it provides interactive control mechanisms for users to split or join clusters of songs they think are (or are not) similar. This feedback not only configures the system but also keeps the user in *control* of the browsing and playlist creation tasks. Users can generate playlists using similarity-based methods, by selecting a cluster from the visualizations.

MuVis [32] is another approach to explore large music collections that supports playlist creation through the use of semantic-ordered treemaps. This approach used content-based similarity between artists (Fluctuation Patterns [77]) for creating the treemaps. MuVis supported playlist creation by either individually selecting each song, or using a similarity-based generation (as explained in Section 3.2.2). The latter approach consisted in two techniques: i) selecting a limited number of songs for a playlist, from the previously filtered songs; ii) a continuous playlist generation using the songs already in the playlist. Torrens et al. [85] also used treemap visualizations to represent music collections, and supported playlist creation by manually selecting and combining regions of interest in the map.

Musicovery[7] is a web-radio that plays songs according to our mood. This solution allows users to select a mood in a 2D space (vertical axis calm-energetic, horizontal axis dark-positive). Playlists are radio-style, where songs are played in sequence based on their similarity. However, no details about similarity are given.

Graphs For a long time, graphs have been studied in mathematics and information technology, and when data is organized in some form of network structure it can be represented as graphs [66]. Representing and accessing music collections using graphs is not an exception.

Crampes et al. [28] presented an innovative approach that makes use of a graph-based visualization to support playlist creation. Their approach is supported through the Multi-Dimensional Scaling projection (MDS) of 46 songs, that is used to scale the solution to thousands of songs. This technique creates *"artistic regions"* in the landscape which can later be used to create adapted and personalized playlists.

Gouyon et al. [43] proposed RAMA, a solution for browsing and exploring music collections through similarity between songs using graphs. Graphs are created using similarity between artists, artist popularity in *Last.fm* and relationships between them. This similarity is represented in the graph by the length of the edges, with longer edges meaning less similarity. Although playlist creation starts with suggestions from the solution, most of it is performed manually by users.

Carreira [21] described an approach for browsing personal music collections using a graph of the similarities between artists, created through the combination of acoustic features with lyrics from the songs. Playlist creation is supported by automatically exploring the similarities between the artists, for example, by selecting one in the graph visualization, or by manually filtering the music collections.

Smarter Playlists[8] is an online web application for supporting the automated creation of playlists using a graph-based visualization. This interactive visualization allows the combination of different sources of songs, like for instance, Artist Radios, Genres or even other playlists, to generate new playlists by defining the steps for their generation (like a script). Moreover, the creators can not only combine different sources of songs, but also apply a set of conditions to specify how song filtering is performed. However, although the structure of the playlist is created manually with creators defining its flow, the final playlist is generated automatically, with no possibility to control the final playlist. Nonetheless, this work stands out as an innovative effort to support playlist creation.

Other techniques: dots and circles, radar Other less typical techniques have also been applied to support interactive playlist creation, from which we highlight the works developed by Goto [41], Vignoli et al. [89], Cardoso et al. [20] and Hilliges et al. [48].

Musicream is a solution for streaming, sticking, sorting, and recalling musical pieces [41]. This technique is based on *random mechanisms* for the displacement of songs, which the authors defend as an interesting approach for promoting serendipitous discovery of

[7] http://musicovery.com/

[8] http://smarterplaylists.playlistmachinery.com/

songs. They implemented the concept of metaplaylist, a playlist composed of several playlists that makes it possible to go back in time to retrieve previous playlists. Playlist generation is supported through similarity-based functions: users start by selecting one song and similar songs are joined automatically to form the playlist.

Vignoli et al. proposed an approach to browse and discover music collections for mobile devices [89]. Rhythm and mood were extracted as main features from acoustic properties to represent the songs in the visualization. The 2D plot can be configured by users, who can select rhythm, mood, genre and year for each one of the axis. Playlist generation is supported by drawing a path over the plot, with the artists (songs) close to the line being added to the playlist.

MOODetector [20] is a solution for automatic playlist generation using mood. The authors extended a typical music player with a mechanism that automatically estimates the arousal and valence values in the Thayer plane (TP). Songs arousal and valence values are calculated using acoustic features. Playlists can be generated using one of three different methods: i) based on a seed song; ii) based on a path drawn by the user; iii) via *“instantaneous search”* (and combined with the previous two).

Hilliges et al. proposed a radar metaphor for visualizing and browsing large music collections called AudioRadar [48]. This solution uses similarity between songs (determined based on content characteristics) and four axis to represent their features: slow-fast, clean-rough, calm-turbulent and melodic-rhythmic. The center of the radar displays the song selected by the user and the remaining ones are placed in the radar accordingly to the similarity values. Playlist creation is performed manually by users, letting them define a range of values for some of the properties of the songs (for example: speed, rhythm and tone), and also their mood.

3.3.6 Advantages and drawbacks

Assisted playlist creation using visualization techniques can combine both manual and automated solutions in a single approach. Not only these visualization techniques can get the best from both types of solutions, but they also are interactive, transparent, promote control and engage users while they are creating playlists.

Nonetheless, visualization techniques can easily become too complex [66], especially when target for non-expert users, like for instance, the end users of streaming services. Therefore, while creating interactive visualizations, researchers should bear in mind the characteristics and limitations of their target users.

3.4 Summary

In this section we approached playlist creation through three perspectives: manual, automatic and assisted. We highlighted the main properties of each approach, detailing relevant related work, and pointing their main advantages and drawbacks.

Moreover, by balancing the benefits and drawbacks of each approach, assisted playlisting techniques appear as the most promising ones for playlist creation, because they can combine both manual and automated solutions in a single approach. Assisted techniques still provide control of playlist creation, possibly requiring less effort and time from creators. However, as far as we known, no thoughtful evaluation or study have been conducted so far to confirm this fact. Testing several visualization techniques with different creation profiles seem a promising path to research.

4 Analysis and discussion

In the previous section we described state-of-the-art research on each different playlist creation method, emphasizing their main advantages and drawbacks. Here we start by analyzing and discussing the methods separately, stressing out how they can complement each other, and finish this section with a comparative analysis between the three.

4.1 Manual playlist creation

Manual creation is the simplest approach for playlist creation and allows users to hold absolute control. Nonetheless, despite its simplicity, selecting and filtering music collections is a hard task and very time consuming, especially because nowadays collections are getting larger [32].

Analysis of current research has evidenced that there is still a need to perform more research to understand current playlist creation habits and behaviors of users [30], particularly while using streaming services that provide access to millions of songs. Furthermore, understanding how users browse and collect songs is also mandatory to develop more adapted solutions. A preliminary study by Hagen [44] presents findings on how users create and manage their playlists in a streaming service. Hagen found out heterogeneous behaviors on the management of the playlists based on *"structural and contextual schemes of aggregating music"*, and also that different levels of user control lead to diverse playlist practices. These practices shed light on new ways of gathering music using streaming services but also demonstrates other techniques that come from pre-digital collecting. Nonetheless, no statistical evidence of these results can be demonstrated, because only 12 participants took part in the study. Further research is required not only to validate these results, but also to develop new solutions that can help playlist creators to easily select songs for their playlists.

Another interesting issue to explore is the behavior of playlist creators, in a similar fashion to what Jennings did for listening habits [51]. As far as we know, no profiling of creation habits and behaviors has been performed. Different profiles could shed some light on different creation habits, useful to adapt and improve future solutions. These profiles could be a valuable asset not only to characterize playlist creators, but also to support personalization of playlist creation solutions, such as, adapting the level of control provided. For example, while some users carefully select each song for a playlist, others might just add some songs from an album or artist. The playlist creation styles described by Lehtiniemi et al. [60] might be a good starting point for future research.

4.2 Automatic playlist creation

Thanks to the increasingly higher computational power available, automatic methods have gained more popularity [14]. Indeed, as we have discussed so far, these methods are useful to promote song discovery and serendipitous discoveries, with little effort [23]. Moreover, these solutions also have the capability of implicitly capturing users' preferences and addressing them individually [34].

Recently, one of the non-traditional methods that have gained importance for handling large data are deep learning techniques [10]. These type of techniques attempt to model high-level abstractions in data by using a deep graph with multiple processing layers. Their main goal is to make better representations and create models to learn these representations from large-scale unlabeled data, through the use of deep neural networks, convolutional deep neural networks, deep belief networks and recurrent neural networks. Until now, and to the best of our knowledge, these techniques have only been applied to music recommendation [63, 86, 91, 92] and not to the subject of playlist generation. Even commercial solutions for music streaming like Spotify[9] have tried these techniques in their environment. Nonetheless, based on the latest trends in music consumption through the use of playlists in streaming services and the access to their large music catalogue, we believe that new playlist generation techniques can benefit from research in deep learning approaches.

Automatic playlisting solutions are only concerned about the algorithms and their optimization, while playlist creators remain apart from this process, having a passive role during creation [78]. With more and more people using streaming services, there are plenty of opportunities to apply and evaluate these techniques in real world scenarios. Furthermore, with millions of users registered in these services, their feedback can help tune the algorithms and improve them with little cost and effort.

Nonetheless, it is important again to emphasize the lack of control of these methods [74]. More research is required to study the application of different levels of control for heterogeneous profiles. For example, creators that crave for more control can benefit from increased customization and parameterizable techniques, while others who need less control, just need a *generate button* that creates playlists with music they would like. Despite the differences of these potential solutions, they would all benefit from including feedback from creators to personalize the suggestions [54, 88].

4.3 Assisted playlist creation

The analysis of the state-of-the-art research suggests that assisted approaches seem to stand out as the best techniques for playlist creation. They can support a fully manual solution, with creators having absolute control over playlist creation, or a completely automatic one, with no control at all. Nonetheless, the analysis performed so far suggests that a more balanced approach seems to be the most promising solution. For example, a solution that allows manual creation with suggestions from automatic techniques could be a mixed approach with the best of both worlds, keeping control but also promoting song discovery and serendipity. Once again, research on different profiles for playlist creation could shed some light on the development of these techniques.

Despite the indicators that suggest that these techniques seem to stand out as the most promising approaches, it is important to notice that further research on the topic is required through several directions, such as, the relationship with different creation profiles, the implementation of different levels of control or the comparison between visualization techniques that support creating the playlists.

In short, the analysis we performed in previous section indicates that: i) maps have been the most used visualization technique for supporting access to music collections, as they are able to represent large sets of data with limited space at the cost of a high level of abstraction [25, 53, 71, 85]; ii) several research works support interaction through path drawing for playlist generation, as a way to engage users in the process of creation [20, 64, 70, 71, 89]; iii) similarity-based techniques (see Section 3.2.2) are the most used for the automatic part of playlist generation, mostly because they can be fully automatized [21, 32, 43]; iv) content (acoustic) features are the most common properties used for generation, mostly because their extraction can also be performed automatically [21, 32, 70, 71, 89].

[9] http://benanne.github.io/2014/08/05/spotify-cnns.html

4.4 Experimental evaluation

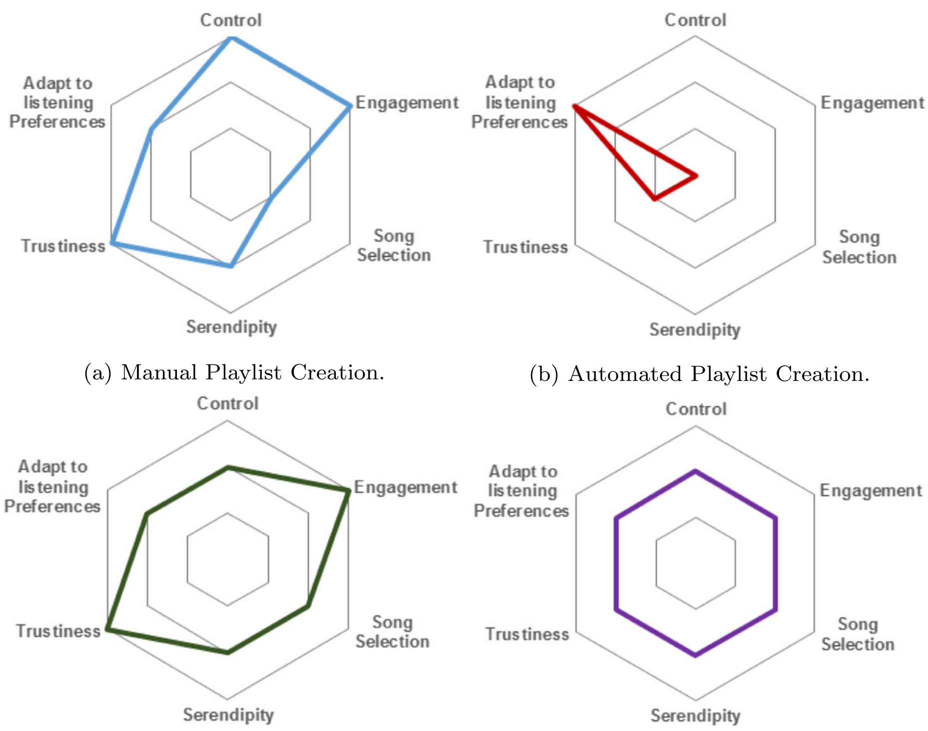
Evaluation in manual techniques is mostly concerned about understanding the creators habits, behaviors and needs for playlist creation [30, 84]. Because manual playlist creation typically involves selecting and picking songs individually for the playlists, evaluation methods do not usually consider playlist quality, but instead they discriminate how creators look for songs, the criteria they use for searching or the contexts for which they create the playlists. Questionnaires, in-person interviews and direct observation are some of the methods employed. Furthermore, sometimes the data collected in these experiments is used as input to train the algorithms in automatic approaches.

For automatic playlist generation techniques, the major goal in the experimental evaluation is to assess the quality of the final playlists generated [14]. To this end, according to both [14] and [67], the quality of the playlists can be measured by performing a subjective evaluation through questionnaires and user feedback, or using a more objective (and automatic) approach, typically based on the analysis of the properties of the songs within the playlists or their sequence. In subjective experiments, participants usually compare and rate the playlists generated by one or more algorithms [7, 75] according to some properties, like for instance, the overall perception of quality and consistency of the playlists, the number of songs that they think that *fit* in the playlist, or the characteristics of the songs that influenced their perception of quality. On one side, these experiments allow developers to understand the perceived quality of the playlists generate by their algorithms regardless how good they perform in terms of objective measures, but on the other side, these techniques are time-consuming and cumbersome, as a high number of participants is required to dilute personal preferences that could masquerade the true results. Usually, objective techniques are cheaper and faster to use, since almost no human intervention is required. Examples of objectives measures include prediction accuracy of measuring how accurately the algorithms can predict the best songs for a playlist given some previous songs [14], or determining quality by measuring a set of features that intrinsically make a good playlist, like the ones identified in previous research: diversity, homogeneity, novelty, freshness, familiarity, smoothness of transitions [67].

Finally, the experimental evaluation of assisted playlisting techniques can usually focus on a two-fold approach [21, 32]: i) measure how effective, efficient and useful the interactive techniques are for playlist creation, and ii) the quality of the final playlists. The same approaches based on user feedback [7, 29, 75] can be applied in these techniques for the two parts. Nonetheless, it is not unusual to rely on automatic metrics as the first step in the evaluation stage.

4.5 Comparison between techniques

The playlist creation process is all about selecting a set of songs to listen to. Besides playing them in sequence or shuffling, we should be able to somehow control the creation process, either by selecting and reordering the songs, filtering out the collection, etc. Overall, we want to keep some variety (or diversity), but maintaining a certain familiarity, while at the same time promoting serendipitous (re)discoveries to listeners (see Fig. 3d for a representation of an ideal method for playlist creation). Regardless of how the approaches we described throughout this paper tackle these issues, they all focus on creating the best playlists that can satisfy the listeners. In Table 6 we compare the three methods presented (manual, automatic and assisted) according to six properties, namely, **Control, Engagement, Trustiness, Song Selection, Serendipity promotion and Adaptability to**



(a) Manual Playlist Creation. (b) Automated Playlist Creation.

(c) Assisted Playlist Creation. (d) Ideal method for playlist creation.

Fig. 3 Analysis of the different methods of playlist creation

the listening preferences, divided in properties for the creator and the listener. These properties were selected based on the discussion performed throughout this survey, both for creators and for listeners.

Manual creation is good because people like and trust handcrafted playlists. It gives control over song selection and ordering. During the creation people are engaged, because users are in charge of all the decisions and actions performed. Despite these benefits, manual

Table 6 Comparison of the different approaches for playlist creation

Target	Properties	Creation technique		
		Manual	Automatic	Interactive
Creators	Control	High	–	Moderate
	Engagement	High	–	High
	Song selection	Hard	–	Moderate
	Serendipity	Moderate	–	Moderate
	Trustiness	High	Low	High
Listeners	Adapt to listening preferences	Moderate	High	Moderate

creation is time-consuming, and song selection from large collections is hard. Even harder to embed or lookup for songs in the long tail, or new releases, because all the decisions and effort is put on the creator (see Fig. 3a for a graphic representation of the Manual Creation method profile).

Automatic creation can easily select and filter songs from large collections (even those from the long tail) usually without any effort from the creators (reason why we do not considered this approaches for analysis in Table 6). They can create playlists suitable for listeners' tastes, as they can easily and without manual interaction model their preferences from different perspectives or facets. However, automatic mechanisms remove all the control from creators, which have been proven to directly decrease listeners' trustiness in the final playlists (see Fig. 3b for the representation of the Automatic Creation method profile).

Assisted techniques are hybrid approaches for playlist creation: not only they engage and give control to users, but they can also assist in the creation by combining manual with automatic methods. Moreover, the automatic component can adapt to users' tastes and promote serendipity (see Fig. 3c). Notice however, that the line between giving some control or full control is thin. Though control is good, too much control can be as bad as no control at all. Perhaps different levels of control should be considered as previously discussed in [40], and once again, concerning the different playlist creation profiles.

5 Conclusions

Music listening has become ubiquitous. Either using personal collections or through streaming services, people can listen to millions of songs almost any time and anywhere without really needing to have those songs. However, with this shift in music consumption and listening, several problems appeared, especially about selecting songs from large music collections to create playlists for the different needs of users. Playlists have been used for several purposes, such as expressing a feeling or conveying a message, joining songs for a certain activity, or even to access such vast music collections. Nonetheless, creating playlists from these large collections also has its challenges.

Playlist creation has been tackled from three different perspectives, namely, manual creation, automated generation and through the use of assisted techniques. In this work we summarized insights from these three perspectives about playlist creation, highlighting their main advantages and drawbacks, and describing how their combination can lead to better playlist creation approaches. This analysis provides evidence that assisted approaches that combine automated with manual techniques stand out as the path to follow for supporting the creation of playlists. They engage users in the creation task, maintaining control and making the process more transparent and enjoyable. However, more research is needed to validate these results.

Regardless of the types of technique used for playlist creation, they all share some characteristics, like for instance, the properties to consider for song selection, song inclusion or exclusion from a playlist, or even the use of visual elements for browsing music collections and supporting playlist creation. We believe that some research is mandatory to summarize such knowledge in a future conceptual framework. Such a framework would not only provide fundamental concepts for future research, but also support the development of more adapted and personalized playlist creation approaches. Furthermore, following experiments should also focus on characterizing playlist creation profiles, to possibly unveil common behaviors, and design a taxonomy for playlist creators.

Acknowledgments This work was supported by national funds through Fundação para a Ciência e Tecnologia, under INESC-ID multiannual funding - PEst-OE/EEI/LA0021/2013 and LaSIGE Strategic Project - UID/CEC/00408/2013. Ricardo Dias was supported by FCT, grant reference SFRH/BD/70939/2010.

References

1. Althoff KD (2001) Case-based reasoning, vol 1. Handbook on software engineering and knowledge engineering
2. Asante MK (2008) It's bigger than hip hop: The rise of the post-hip-hop generation Macmillan
3. Aucouturier JJ, Pachet F (2002) Scaling up music playlist generation. In: Proceedings of the IEEE international conference on multimedia and expo, vol 1, pp 105–108
4. Baccigalupo C, Plaza E (2006) Case-based sequential ordering of songs for playlist recommendation. In: Advances in case-based reasoning, lecture notes in computer science, vol 4106. Springer, Berlin Heidelberg, pp 286–300
5. Baccigalupo C, Plaza E (2007) A case-based song scheduler for group customised radio. In: Case-based reasoning research and development, vol 4626. Springer Verlag, pp 433–448
6. Bakalov F, Meurs MJ, König-Ries B., Sateli B, Witte R, Butler G, Tsang A (2013) An approach to controlling user models and personalization effects in recommender systems. In: Proceedings of the international conference on intelligent user interfaces, pp 49–56
7. Barrington L, Oda R, Lanckriet G (2009) Smarter than genius? human evaluation of music recommender systems. In: ISMIR, vol 9. Citeseer, pp 357–362
8. Baur D, Seiffert F, Sedlmair M, Boring S (2010) The streams of our lives: Visualizing listening histories in context. IEEE TVCG
9. Baur D, Hering B, Boring S, Butz A (2011) Who needs interaction anyway: exploring mobile playlist creation from manual to automatic. In: Proceedings of the international conference on intelligent user interfaces
10. Bengio IGY, Courville A Deep learning (2016). http://www.deeplearningbook.org. Book in preparation for MIT Press
11. Bennett J, Lanning S, Netflix N (2007) The netflix prize. In: KDD cup and workshop in conjunction with KDD
12. Bonnin G, Jannach D (2013) A comparison of playlist generation strategies for music recommendation and a new baseline scheme. In: Workshops at the twenty-seventh AAAI conference on artificial intelligence
13. Bonnin G, Jannach D (2013) Evaluating the quality of generated playlists based on hand-crafted samples. In: Proceedings of the international music information retrieval conference
14. Bonnin G, Jannach D (2014) Automated generation of music playlists: Survey and experiments. ACM Comput Surv 47(2):26
15. Bostandjiev S, O'Donovan J, Höllerer T (2012) Tasteweights: A visual interactive hybrid recommender system. In: Proceedings of the sixth ACM conference on recommender systems, RecSys '12. ACM, NY, USA, pp 35–42
16. Brewster B, Broughton F (2006) Last night a DJ saved my life: The history of the disc jockey, 2nd edn. Headline Book Publishing, London, United Kingdom
17. Bull M (2006) Investigating the culture of mobile listening: From walkman to ipod. In: Consuming Music Together. Springer Verlag, pp 131–149
18. Burke R (2002) Hybrid recommender systems: Survey and experiments. User Model User-Adapt Interact 12(4):331–370
19. Byron L, Wattenberg M (2008) Stacked graphs geometry & aesthetics. IEEE TVCG. doi:10.1109/TVCG.2008.166
20. Cardoso L, Panda R, Paiva RP (2011) Moodetector: A prototype software tool for mood-based playlist generation. In: Simposio de Informatica–INForum 2011, vol 124
21. Carreira M (2012) Musicmapper: Visualizing and exploring music libraries. Master's Thesis, Instituto Superior Tecnico, Universidade de Lisboa
22. Casey M, Veltkamp R, Goto M, Leman M, Rhodes C, Slaney M (2008) Content-based music information retrieval: current directions and future challenges. Proc IEEE 96(4):668–696
23. Celma O (2010) Music Recommendation and Discovery. Springer, Berlin Heidelberg
24. Chen Y (2010) Exploratory browsing: Enhancing the browsing experience with media collections. Master's Thesis, Ludwig-Maximilians-Universitt Mnchen

25. Chen YX, Butz A (2009) Musicsim: Integrating audio analysis and user feedback in an interactive music browsing ui. In: Proceedings of the international conference on intelligent user interfaces, pp 429–434
26. Chen S, Moore JL, Turnbull D, Joachims T (2012) Playlist prediction via metric embedding. In: Proceedings of the international conference on knowledge discovery and data mining. ACM, pp 714–722
27. Cohen WW, Fan W (2000) Web-collaborative filtering: recommending music by crawling the Web. Comput Netw 33(1):685–698
28. Crampes M, Villerd J, Emery A, Ranwez S (2007) Automatic playlist composition in a dynamic music landscape. In: Proceedings of the international workshop on semantically aware document processing and indexing, pp 15–20
29. Cremonesi P, Garzotto F, Negro S, Papadopoulos AV, Turrin R (2011) Looking for good? recommendations: A comparative evaluation of recommender systems. In: Human-computer interaction–INTERACT 2011. Springer, pp 152–168
30. Cunningham SJ, Bainbridge D, Falconer A (2006) More of an art than a science: Supporting the creation of playlists and mixes. In: Proceedings of 7th international conference on music information retrieval, Victoria, Canada, pp 240–245
31. De Mooij A (1997) Learning preferences for music playlists. Master's Thesis, Technische Universiteit Eindhoven, Department of Mathmatics and Computer Science
32. Dias R, Fonseca MJ (2010) Muvis: an application for interactive exploration of large music collections. In: Proceedings of the international conference on multimedia (MM'10). ACM, pp 1043–1046
33. Dias R, Fonseca MJ, Gonalves D (2012) Interactive exploration of music listening histories. In: Proceedings of the international conference on Advanced Visual Interfaces, AVI'12. ACM, NY, USA
34. Dias R, Cunha R, Fonseca MJ (2014) A user-centered music recommendation approach for daily activities. In: Proceedings of the workshop on new trends in content-based recommender systems (CBRecSys'14)
35. Dix A, Ellis G (1998) Starting simple: Adding value to static visualisation through simple interaction. In: Proceedings of the working conference on advanced visual interfaces, AVI'98, pp 124–134
36. Du Gay P, Hall S, Janes L, Madsen AK, Mackay H, Negus K (2013) Doing cultural studies: The story of the Sony Walkman. Sage
37. Fields B (2011) Contextualize your listening: The playlist as recommendation engine. Ph.D. Thesis, Goldsmiths University of London
38. Fields B, Lamere P (2010) Finding a path through the jukebox the playlist tutorial. In: International conference on music information retrieval 2010
39. Freire AM (2008) Remediating radio: Audio streaming, music recommendation and the discourse of radioness. Intl Stud Broadcast Audio Media 5(2-3):97–112
40. Golbeck J, Hansen DL (2011) A framework for recommending collections. In: Inproceedings of the workshop on novelty and diversity in recommender systems (DiveRS'11), p 35
41. Goto M, Goto T (2005) Musicream: New music playback interface for streaming, sticking, sorting, and recalling musical pieces. In: Inproceedings of the international conference on music information retrieval, pp 404–411
42. Goussevskaia O, Kuhn M, Wattenhofer R (2008) Exploring music collections on mobile devices. In: Proceedings of the international conference on human computer interaction with mobile devices and services, pp 359–362
43. Gouyon F, Cruz N, Sarmento L (2011) A last.fm and youtube mash-up for music browsing and playlist edition. In: Late-breaking demo session at the international music information retrieval conference. http://ismir2011.ismir.net/latebreaking/LB-3.pdf
44. Hagen AN (2015) The playlist experience: Personal playlists in music streaming services. Popular Music and Society (ahead-of-print):1–21
45. Hariri N, Mobasher B, Burke R (2012) Context-aware music recommendation based on latent-topic sequential patterns. In: Proceedings of the international conference on recommender systems (RecSys'12), pp 131–138. doi:10.1145/2365952.2365979
46. Heise S, Hlatky M, Loviscach J (2008) Soundtorch: quick browsing in large audio collections. In: Proceedings of the convention of the audio engineering society
47. Herrera P (2010) Rocking around the clock eight days a week: an exploration of temporal patterns of music listening. In: 1st Workshop On music recommendation and discovery (WOMRAD). ACM RecSys, pp 7–10
48. Hilliges O, Holzer P, Klber R, Butz A (2006) Audioradar: A metaphorical visualization for the navigation of large music collections. In: Smart graphics, lecture notes in computer science, vol 4073. Springer, Berlin Heidelberg, pp 82–92
49. Hsu JL, Chung SC (2011) Constraint-based playlist generation by applying genetic algorithm. In: IEEE international conference on systems, man, and cybernetics (SMC). IEEE, pp 1417–1422

50. Jannach D, Kamehkhosh I, Bonnin G (2014) Analyzing the characteristics of shared playlists for music recommendation. In: Proceedings of the 6th workshop on recommender systems and the social web
51. Jennings D (2007) Blogs and Rock 'n' Roll: How Digital Discovery Works and what it Means for Consumers, Creators and Culture. Nicholas Brealey Publication
52. Kamalzadeh M, Baur D, Möller T (2012) A survey on music listening and management behaviours. In: Proceedings of the international conference on music information retrieval, pp 373–378
53. Knees P, Schedl M, Pohle T, Widmer G (2006) An innovative three-dimensional user interface for exploring music collections enriched with meta-information from the web. In: Proceedings of the 14th international conference on multimedia (MM'06). ACM Press, pp 17–24
54. Knijnenburg BP, Bostandjiev S, O'Donovan J, Kobsa A (2012) Inspectability and control in social recommenders. In: Proceedings of the 6th conference on recommender systems (RecSys'12), pp 43–50
55. Kohonen T (1982) Self-organized formation of topologically correct feature maps. Biol Cybern 43(1):59–69
56. Konstan JA, Miller BN, Maltz D, Herlocker JL, Gordon LR, Riedl J, Volume H (1997) Grouplens: Applying collaborative filtering to usenet news. Commun ACM 40:77–87
57. Kosara R, Hauser H, Gresh DL (2003) An interaction view on information visualization. state-of-the-art report. In: Proceedings of EUROGRAPHICS
58. Kremp PA (2010) Innovation and selection: symphony orchestras and the construction of the musical canon in the United States (1879–1959). Soc Forces 88(3):1051–1082
59. Laplante A, Downie JS (2006) Everyday life music information-seeking behaviour of young adults. In: Proceedings of the international conference on music information retrieval, pp 381–382
60. Lehtiniemi A, Seppänen J (2007) Evaluation of automatic mobile playlist generator. In: Proceedings of the 4th international conference on mobile technology, applications, and systems and the 1st international symposium on computer human interaction in mobile technology, mobility '07, pp 452–459
61. Leitich S, Topf M (2007) Globe of music - music library visualization using geosom. In: Dixon S., Bainbridge D., Typke R. (eds) Proceedings of the international conference on music information retrieval, pp 167–170
62. Leong T, Howard S, Frank V (2008) Choice: Abdicating or exercising. In: Proceedings of the 26th annual chi conference on human factors in computing systems
63. Liang D, Zhan M, Ellis DPW (2015) Content-aware collaborative music recommendation using pre-trained neural networks. In: International conference on music information retrieval
64. Lillie AS (2008) Musicbox: Navigating the space of your music. Master's Thesis Massachussets Institute of Technology
65. Marchionini G (2006) Exploratory search: From finding to understanding. Commun ACM 49(4):41–46
66. Mazza R (2009) Introduction to information visualization. Springer Science & Business Media
67. Mcfee B, Lanckriet G (2011) The natural language of playlists. In: Proceedings of the international society for music information retrieval conference, pp 537–541
68. Miller S, Reimer P, Ness SR, Tzanetakis G (2010) Geoshuffle: Location-aware, content-based music browsing using self-organizing tag clouds. In: Proceedings of the international conference on music information retrieval, pp 237–242
69. Morchen F, Ultsch A, Ncker M, Stamm C (2005) Databionic visualization of music collections according to perceptual distance. In: Proceedings of the international conference on music information retrieval, pp 396–403
70. Neumayer R, Dittenbach M, Rauber A (2005) Playsom and pocketsomplayer, alternative interfaces to large music collections. In: Proceedings of the international conference on music information retrieval
71. Pampalk E, Rauber A, Merkl D (2002) Content-based organization and visualization of music archives. In: Proceedings of the 10th ACM international conference on multimedia (MM'02), pp 570–579
72. Pampalk E, Rauber A, Merkl D (2002) Using smoothed data histograms for cluster visualization in self-organizing maps. Springer
73. Parra D (2012) Beyond lists: Studying the effect of different recommendation visualizations. In: Proceedings of the sixth ACM conference on recommender systems, RecSys '12. ACM, NY, USA, pp 333–336. doi:10.1145/2365952.2366035
74. Parra D (2013) User controllability in a hybrid recommender system. Ph.D. Thesis, University of Pittsburgh
75. Pauws S, Eggen B (2002) Pats: Realization and user evaluation of an automatic playlist generator. In: Proceedings of the international conference on music information retrieval
76. Pauws S, Verhaegh W, Vossen M (2008) Music playlist generation by adapted simulated annealing. Inf Sci 178(3):647–662

77. Schedl M, Knees P, Widmer G (2005) Interactive poster: using comirva for visualizing similarities between music artists. In: Proceedings of the 16th IEEE visualization 2005 conference (Vis'05), MN, USA
78. Schedl M, Flexer A, Urbano J (2013) The neglected user in music information retrieval research. J Intell Inf Syst 41(3):523–539
79. Schwartz B (2005) The paradox of choice: why more is less harper perennial
80. Shneiderman B (1992) Tree visualization with tree-maps: 2-d space-filling approach. ACM Trans Graph 11(1):92–99
81. Siegel D (2008) Mindsight. Oneworld, Oxford
82. Slaney M, White W (2007) Similarity based on rating data. In: Proceedings of the international conference on music information retrieval
83. Sneha Antony JJN (2014) Survey on playlist generation techniques. Int J Adv Res Comput Eng Technol (IJARCET) 3:437–439
84. Stumpf S, Muscroft S (2011) When users generate music playlists: When words leave off, music begins? In: Third international workshop on advances in music information research (AdMIRe)
85. Torrens M, lluis Arcos J (2004) Visualizing and exploring personal music libraries. In: Proceedings of the international conference on multimedia information retrieval
86. Van den Oord A, Dieleman S, Schrauwen B (2013) Deep content-based music recommendation. In: Burges C. J. C., Bottou L., Welling M., Ghahramani Z., Weinberger K. Q. (eds) Advances in neural information processing systems 26. Curran Associates, Inc, pp 2643–2651
87. Verbert K, Parra D, Brusilovsky P, Duval E (2013) Visualizing recommendations to support exploration, transparency and controllability. In: Proceedings of the international conference on intelligent user interfaces, pp 351–362
88. Vignoli F, Pauws S (2005) A music retrieval system based on user driven similarity and its evaluation. In: Proceedings of the international conference on music information retrieval, pp 272–279
89. Vignoli F, Van Gulik R, Van de Wetering H (2004) Mapping music in the palm of your hand, explore and discover your collection. In: Proceedings of the international conference on music information retrieval
90. Wall T (2007) Finding an alternative: Music programming in us college radio. Radio J: Int Stud Broadcast Audio Media 5(1):35–54
91. Wang X, Wang Y (2014) Improving content-based and hybrid music recommendation using deep learning. In: Proceedings of the 22Nd ACM international conference on multimedia, MM '14
92. Wang H, Wang N, Yeung DY (2015) Collaborative deep learning for recommender systems. In: Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '15
93. Weigl DM, Guastavino C (2011) User studies in the music information retrieval literature. In: Proceedings of the international conference on music information retrieval

Ricardo Dias is a PhD student at the Computer Science and Engineering Department at IST/ULisbon, Portugal. He is a researcher at INESC-ID Visualization and Intelligent Multimodal Interfaces Group since 2010. He participated in two National projects (ColaDI and Crush). His research interests are in Playlist Creation Techniques, Music Visualization and Retrieval, and Music Recommendation.

Daniel Gonçalves is a researcher at the Visualization and Multimodal Interfaces Group INESC-ID and Professor of Computer Science at Instituto Superior Técnico (IST/UL), Portugal. His research interests include the areas of Information Visualization, Personal Information Management, Human-Computer Interaction and Accessibility. He is a member of ACM and the Portuguese Computer Graphics Group (the national Eurographics chapter).

Manuel J. Fonseca holds a PhD (2004) in Information Systems and Computer Engineering from IST/ULisbon, is an Associated Professor at Faculty of Sciences, University of Lisbon, and a senior researcher at the Human-Computer Interaction and Multimedia group of LaSIGE. His main research areas include Human-Computer Interaction, Brain-Computer Interfaces (BCI), Emotions Recognition, Multimedia Information Retrieval, Sketch Recognition, and Health Systems. He is a senior member of IEEE and ACM.