ORIGINAL RESEARCH

Exploring playlist titles for cold-start music recommendation: an effectiveness analysis

Ali Yürekli[1] · Alper Bilge[2] · Cihan Kaleli[1]

Abstract

In music recommender systems, automatic playlist continuation is an emerging task that aims to improve users' listening experience by recommending music in line with their musical taste. The typical approach towards this goal is to identify playlist characteristics by inspecting the existing tracks (i.e., seeds) in target playlists. However, seeds are not always available, especially when users create new playlists. For such cold-start situations, user-generated titles can be a good starting point to understand the intended purpose of users. This paper investigates the effectiveness of titles as an auxiliary data source for playlists suffering from the cold-start problem. Employing three naive recommendation models, we conduct experiments on one million music playlists from the Spotify platform. Our analyses show that the prevalent attitude in naming playlists results in highly accurate recommendations for playlists concerning a specific theme, such as albums, artists, and soundtracks. As the title space moves away from a particular theme, recommendation accuracy drops. Furthermore, the correlation between the common preference of a title and its usability in recommendation is quite weak; a title without a common sense may be useless in recommender systems, even though many users favor that title. Consequently, our findings serve as a guideline to develop title-aware recommendation approaches that can provide coherent continuations to the cold-start playlists.

Keywords Music recommender systems · Automatic playlist continuation · Cold-start problem · Playlist titles · User-generated content

1 Introduction

The playlist concept in music has become an essential utility in almost all leading music streaming platforms like Spotify[1] or Apple Music.[2] In simple words, a playlist is a sequence of tracks meant to be listened to together (Schedl et al. 2018). As the definition suggests, playlists fit quite well with the sequential consumption of music and offer a continuous listening experience to the users. At this point, music recommender systems (MRS) come into play to retain

the experience in a personalized manner. Recommendations produced by MRS should adapt to the dynamics of the target playlist, which requires a well-understanding and effective modeling of music playlists (Vall et al. 2019). This task, the extension of listening sessions in a coherent way that fits the original characteristics of playlists, is referred to as automatic playlist continuation (APC) and considered one of the grand challenges in MRS research (Schedl et al. 2018).

In APC, the main objective is to infer the intended purpose of a playlist. Given background information (e.g., user profiles, previous playlists, or listening logs) on a music collection, characteristics of a target playlist can be understood to a certain extent (Bonnin and Jannach 2014). This way, it is possible to find *good recommendations* that can continue music playlists. However, in this context, the perception of goodness is highly subjective. In a qualitative user study, Lee (2011) exhibits that user perceptions of new music are strongly affected by various factors, including personal preferences, similarity, variety, and familiarity. In addition to

✉ Cihan Kaleli
  ckaleli@eskisehir.edu.tr

  Ali Yürekli
  aliyurekli@eskisehir.edu.tr

  Alper Bilge
  abilge@akdeniz.edu.tr

[1] Department of Computer Engineering, Eskişehir Technical University, 26555 Eskişehir, Turkey

[2] Department of Computer Engineering, Akdeniz University, 07070 Antalya, Turkey

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

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

this subjectivity, temporal dynamics, such as mood, context, and emotional state, should be considered when recommending new music to users (Gillhofer and Schedl 2015). Furthermore, music catalogs are massive in size; the item space is much larger and diverse in music than other recommender systems (e.g., movie and book recommenders). For all these grounds, APC can be stated as a compelling task in the MRS field.

Although APC is challenging in itself (due to several factors, some of which are mentioned above), the task gets even more complicated for the newly created playlists. The cold-start problem, a significant issue of recommender systems in general (Elahi et al. 2016), also arises in APC in the absence of information about user or playlist characteristics.

1.1 Problem definition

In a typical recommender system, the cold-start problem occurs due to the lack of predictive data for new users or items (Bobadilla et al. 2012). When a new user registers to a system, she does not have personal preference indicators (e.g., ratings and votes) and cannot receive personalized recommendations, resulting in the cold-start user problem (Rashid et al. 2008). Similarly, when a new item enters a product catalog, the system is unaware of the item in terms of historical user actions, which leads to the cold-start item problem (Park and Tuzhilin 2008). In terms of the APC task in MRS, playlists and tracks can be regarded as users and items, respectively. In line with this analogy, one can discuss the cold-start APC situation in two cases: (i) cold-start playlist problem and (ii) cold-start track problem. During this study, we discuss the first cold-start case.

When a user creates a fresh music playlist (initially without any tracks), there exist no clues indicating characteristics of that playlist. Consequently, MRS have difficulties in performing predictive analyses for next track recommendations. In such cases, playlist titles might be a possible starting point to create models that can alleviate this cold-start problem. Intuitively, playlist titles, which are user-generated namings, are expected to carry information about the intended purpose of corresponding playlists. The inference attained by these titles can exist in many different forms, such as musical genre, artist name, date of playlist creation, and user mood (Schedl et al. 2015). On the other hand, some playlist titles (e.g., 'my music', 'favorites', and 'for me') may not contain any semantic information or be too generic to give clues about user intent (Faggioli et al. 2018). Although several studies have utilized titles for cold-start APC, it has not been explored extensively what types of titles users create and which of these types are useful in cold-start solutions.

1.2 Contributions and organization

In this work, we investigate the effectiveness of utilizing playlist titles as additional information to deal with the cold-start APC, in which no seed tracks are provided. In our analyses, we use the Million Playlist Dataset[3] (MPD), a collection of one million user-generated playlists from the Spotify platform. Reproducing the cold-start scenario for all playlists in the MPD, we measure and compare recommendation performances of three approaches based on track popularity, title matching, and title proximity. Then, we identify and categorize the content in playlist titles that result in highly accurate recommendations when no other information is available. We believe that our findings can expand horizons in MRS research and help researchers to develop effective models to alleviate the cold-start problem in APC. In brief, the main contributions of the work can be listed as follows:

1. The effectiveness of title information in cold-start APC solutions is analyzed on a large scale, containing one million test instances.
2. The significance of titles in recommending proper tracks to continue cold-start playlists is determined, and highly effective titles are thematically categorized.
3. A guideline to develop title-aware recommendation approaches for cold-start APC is presented.

The remainder of the paper is organized as follows: Sect. 2 explains how current studies in the MRS literature use playlist titles for music recommendations. Sect. 3 describes the recommendation models employed during the experiments. In Sect. 4, the dataset explored in this study, the perspective on the cold-start problem, and evaluation metrics are introduced. Subsequently, Sect. 5 presents the experimental results and corresponding analyses. In Sect. 6, special notes and discussion of our findings are included. Finally, conclusions are drawn and future work is pointed out in Sect. 7.

2 Related work

Utilizing additional information is a common strategy to deal with the cold-start problem in recommender systems (Bobadilla et al. 2012). In terms of music recommendations, an auxiliary data source might help understand user intent, which is a key factor in improving recommendation quality (Schedl et al. 2013) and alleviating the cold-start problem. This section presents how recent studies in the MRS literature use playlist titles for recommendation purposes.

---

[3] https://recsys-challenge.spotify.com.

The music preferences of users vary according to contextual factors such as time, location, and mood. For example, a person's music choice when working at the office on a Monday morning may differ from the music she listens to when resting at home on a Sunday evening. Accordingly, Schedl et al. (2013) express user context as dynamic and frequently changing conditions. The authors categorize the factors influencing human music perception and promote user context as a major factor along with music content (e.g., timbre and melody), music context (e.g., semantic labels and lyrics), and user properties (e.g., demographics and musical experience). Especially in the absence of explicit information about user context, playlist titles might be utilized as indicators of user needs (Schedl et al. 2018).

Several approaches have been proposed to make use of the latent factors between user context and playlist titles. Pichl et al. (2015) combine the #nowplaying dataset by Zangerle et al. (2014) with playlist titles crawled from Spotify, which are then aggregated into contextual clusters (e.g., 'summer' cluster contains titles such as 'my summer playlist', 'hot outside', and 'summer 2015 tracks'). The authors incorporate the attained contextual information into a collaborative filtering system and observe at least 33% performance boost in precision. In a complementary research (Pichl et al. 2017), the same authors use factorization machines to extract information about user context from the playlist titles. This way, the drawbacks of contextual pre-filtering are avoided as factorization machines directly incorporate the contextual clusters into the music recommendation process. Chung et al. (2017) propose an unsupervised learning method to represent music tracks and words extracted from playlist titles in a common latent space. Casting a text-based music retrieval task into a nearest neighbor search problem, the authors examine thematic inferences of the learned representations for music queries such as 'christmas', 'punk', and 'coldplay'.

2.1 Cold-start approaches in the RecSys Challenge 2018

The RecSys Challenge 2018 (Chen et al. 2018), focusing on the task of APC, has drawn significant attention to the research on the subject. Briefly, the main goal of the challenge is to build a recommendation model that can add more tracks (fixed to 500 tracks for the challenge) to existing playlists. The proposed models are evaluated on a test collection that covers ten different categories, one of which is related to the cold-start problem. When we analyze the approaches proposed for this category, we observe two main strategies adopted by the participants: (i) developing a problem-specific solution or (ii) adapting an approach, which is proposed for other categories, to process playlist titles.

Table 1 Techniques applied by the top-performing approaches of the RecSys Challenge 2018 for the cold-start scenario

Publication	SM	MF	NN	CBF	NNS
Volkovs et al. (2018)	✓	✓	✗	✗	✗
Yang et al. (2018)	✓	✗	✓	✗	✗
Rubtsov et al. (2018)	✓	✓	✗	✗	✗
Antenucci et al. (2018)	✓	✗	✗	✓	✗
Zhu et al. (2018)	✓	✓	✗	✗	✓
Ludewig et al. (2018)	✓	✓	✗	✗	✗
Kallumadi et al. (2018)	✗	✗	✗	✗	✓
Kelen et al. (2018)	✗	✗	✗	✗	✓
Faggioli et al. (2018)	✓	✗	✗	✓	✗
Ferraro et al. (2018)	✗	✓	✗	✗	✗
Zhao et al. (2018)	✓	✗	✗	✓	✗
Kaya and Bridge (2018)	✓	✗	✗	✓	✗
Kim et al. (2018)	✗	✓	✓	✗	✗

SM, MF, NN, CBF, and NNS denote separate model, matrix factorization, neural networks, content-based filtering, and nearest neighbor search, respectively. The table is inspired from Zamani et al. (2019)

In Table 1, we provide an overview of the cold-start techniques applied by top-performing approaches of the challenge. As the table presents, the majority of the participants prefer the first strategy and build a separate model specific to the problem. The reason for this preference is that even a single track is fairly informative about the rest of the playlist (Kelen et al. 2018); the lack of seed tracks raises the need for separate solutions that can utilize other available information (i.e., playlist titles in this case). Furthermore, the techniques in these approaches compose of matrix factorization, neural networks, content-based filtering, and nearest neighbor search. Among all, matrix factorization on top of playlist titles is the most preferred way to deal with cold-start APC.

Recommendation models use matrix factorization to decompose user-item matrices into lower dimensions (Koren et al. 2009). In cold-start APC, the construction of this user-item matrix is commonly achieved by treating the unique titles as users, the tracks as items, and the number of co-occurrences as ratings (Ludewig et al. 2018; Zhu et al. 2018; Rubtsov et al. 2018; Ferraro et al. 2018; Volkovs et al. 2018). Despite minor differences in handling titles (e.g., the usage of top-2000 most frequent words (Rubtsov et al. 2018)), matrix factorization can successfully extract latent factors to find similar titles in terms of the tracks contained in playlists. Remarkably, for each evaluation metric that assesses the recommendation quality of proposed models (explained later in Sect. 4.3), a different matrix factorization technique (i.e., Name-SVD (Volkovs et al. 2018), CS_{text} (Rubtsov et al. 2018), and Title-MF (Ludewig et al. 2018)) achieves the best performance, as shown in Table 2.

Within the scope of the RecSys Challenge 2018, several techniques other than matrix factorization have also been

Table 2 The models achieving highest performance in the cold-start category of the RecSys Challenge 2018

Model	R-precision@500	NDCG@500	RSC@500
Name-SVD	**0.0978**	0.2044	10.759
CS_{text}	0.0845	0.1881	**10.423**
Title-MF	0.0953	**0.2053**	10.563

Complete list of the performances of top-performing approaches can be found in Zamani et al. (2019)

utilized to discover relationships between titles and tracks. In neural network approaches, Yang et al. (2018) and Kim et al. (2018) propose character-level implementations of convolutional and recurrent neural nets to learn title representations, respectively. The reason behind the preference of character-level embedding over word-level embedding is out-of-dictionary words and typos in titles. Some content-based approaches use title tokens and normalized titles to build contextual clusters (Antenucci et al. 2018; Faggioli et al. 2018; Kaya and Bridge 2018; Zhao et al. 2018). Differently, Kallumadi et al. (2018) approach the problem as an ad-hoc retrieval task and propose an ensemble of query expansion and nearest neighbor search on playlist metadata.

When categorical performances of top-performing approaches are compared, it is seen that the cold-start category is the one with the lowest scores Zamani et al. (2019). Furthermore, the set of performance scores in this category can be considered quite uniform. Despite the utilization of several state-of-the-art techniques, including deep neural networks (Kim et al. 2018; Yang et al. 2018), most cold-start APC models in the challenge perform closely and cannot efficiently explore playlist titles. Zamani et al. (2019) associate this shortage of well-understanding of playlist titles with the sparseness of titles (92,944 unique titles out of one million playlists) and the scale of the MPD. Agreeing with this reasoning, we further claim that playlist titles might not be deeply analyzed on purpose. Since the challenge event is a time-limited competition rather than academic research, the participants might inherently prefer to focus on seed-based tasks (i.e., nine categories that constitute 90% of the overall score of a participant).

The abovementioned cold-start APC approaches in the literature quickly focused on producing recommendations in a competition theme rather than deep understanding and analysis of the title content. On the other hand, we focus on what kind of information titles contain, which titles are effective in understanding the purpose of playlists, and how a title-aware recommendation approach should be formed. Thus, our work differs from current studies in these aspects and plays a guiding role in the cold-start problem in MRS research.

3 Recommendation models

This section presents the recommendation models used to assess how suitable and effective playlist titles are in the cold-start APC problem. The proposed models are straightforward; they represent three basic ideas about the usage of titles when addressing the problem: (i) not including any title information in recommendation, (ii) using titles as exactly they are, and (iii) normalizing titles into a plain form.

3.1 Global track popularity

A simple solution to cold-start APC problem could be replenishing new playlists with popular tracks among other playlists. This approach does not utilize any title information; it is solely based on track occurrences in the dataset. Let P be the set of all playlists in the MPD, a global popularity score is calculated for each unique track $t \in P$, as shown in Eq. 1. Then, tracks are sorted by descending order of their popularity scores. Top-n tracks of the sorted collection constitute the recommendations for cold-start playlists.

$$popularity(t) = \sum_{p \in P} \mathbb{1}(t, p) \quad \mathbb{1}(t, p) = \begin{cases} 1 & \text{if } t \in p \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Due to its simplicity, several approaches in RecSys Challenge 2018 (Kaya and Bridge 2018; Faggioli et al. 2018; Ludewig et al. 2018; Ferraro et al. 2018) have adopted global popularity as a complementary method for the cases in which an adequate number of recommendations could not be reached (each playlist must be provided with 500 recommended tracks in the challenge). However, the recommendation accuracy of this method on its own has not been explored yet.

3.2 Title matching

User-generated metadata, such as titles and descriptions, can be utilized to compensate the lack of explicit information about playlist characteristics (Schedl et al. 2018). Assuming that playlists with identical titles are likely to share common user intent, a recommendation model based on title matching can be proposed to alleviate the cold-start APC problem. Supporting this assumption, there exist 92,944 unique titles among one million playlists of the MPD, as shown in Table 3.

Let P and i be the set of all playlists in the MPD and target playlist to suggest recommendations, respectively. For each unique track $t \in P$, a relevance score is calculated as shown in Eq. 2. Differently from global track popularity, this measure incorporates title matching into relevancy estimation. The recommendation process is carried out by selecting

Some Spellings of Country

Some Spellings of Summer 2017

Fig. 1 Two examples illustrating how users spell same titles differently

top-*n* tracks from the ranking according to the highest relevance scores.

$$relevance(t) = \sum_{p \in P} \mathbb{1}(t, p, i) \quad \mathbb{1}(t, p, i) = \begin{cases} 1 & \text{if } t \in p \wedge \text{title}_p = \text{title}_i \\ 0 & \text{otherwise} \end{cases}$$

(2)

3.3 Normalized title matching

In social media, users typically do not conform to language rules and prefer an informal writing style, which leads to spelling errors, acronyms, out-of-dictionary slang, intentional misspelling, and phonetic spelling in user-generated content (Clark and Araki 2011). This fact is also true for music playlist titles. Users express the same terms in many different ways. Figure 1 presents two examples of how users state the same titles with different spellings. However, it is possible to homogenize playlist titles and obtain a condensed title set by applying text normalization on the content.

The MPD comes up with a basic normalization method that lowercases all letters, trims whitespaces, and removes some special characters from the text. When applied on one million titles in the dataset, this method results in a total number of 17,381 unique normalized titles, each of which can be considered as a textual cluster. A recommendation approach based on these clusters can be used to come up with candidate tracks for playlists suffering from the cold-start problem. Initially, all titles in the MPD should be transformed into plain forms by text normalization. Then, for a target playlist *i*, the corresponding textual cluster is retrieved by the normalized title of *i*. Using the above Eq. 2 with normalized titles, the relevance scores of tracks within cluster

are calculated. Regarding track relevancy, top-*n* tracks form the list of recommedations for *i*.

The technique used to normalize titles directly affects the recommendation performance of the proposed method. Applying natural language processing (NLP) techniques such as stemming and lemmatization (Zamani et al. 2019), or contextual clustering (Pichl et al. 2015) may result in better understanding of latent factors between titles and tracks. However, we prefer to use the aforementioned standard normalization procedure provided within the MPD in order to ensure reproducible results and analyses on title effectiveness for the cold-start problem.

4 Experimental setup

In this section, we describe our experimental setup and methodology. Firstly, we introduce the dataset used in our experiments by providing basic demographics and statistics. Then, we describe how we simulate the cold-start problem in APC for all playlist instances in the dataset. Finally, we present the evaluation metrics used to assess the ability of recommendation models.

4.1 The million playlist dataset

During the study, we conduct experiments on the MPD, which is an extensive music collection initially released for the RecSys Challenge 2018. The dataset contains one million user-generated playlists created by the users who reside in the United States and are at least 13 years old.

A playlist instance in the MPD contains a title, an optional description, date of modification, a list of tracks

with track metadata (artist, album, and track information), and some other miscellaneous fields about the playlist. As illustrated in Table 3, the MPD comprises more than two million unique tracks associated with hundreds of thousands of unique artists and albums. Including such enriched music music content from the Spotify platform, the MPD offers an excellent research opportunity in the field of MRS.

4.2 Simulating the cold-start problem

In recommender systems, the cold-start problem can be encountered on the basis of new users or new products (Rashid et al. 2008; Park and Tuzhilin 2008). This study focuses on the first case and investigates the effectiveness of playlist titles as an additional information source to address the cold-start playlist problem.

In the cold-start scenario of the RecSys Challenge 2018, the participants are expected to produce recommendations for a small collection of 1000 playlists with corresponding titles, in which all tracks are withheld (Chen et al. 2018). In other words, these test instances do not contain any seed tracks; the only information available for a playlist is the title specified by the owner of that playlist. When simulating the problem in our experiments, we adopt the same approach. However, we prefer to examine every instance in the MPD for a comprehensive analysis. Since all playlists in the dataset have a title and at least one track, the cold-start scenario can be applied to every instance. Thus, we follow a testing strategy similar to leave-one-out cross-validation (LOOCV), in which recommendation models are built on the whole dataset, but copied and updated for each recommendation request, as illustrated in Fig. 2.

4.3 Evaluation metrics

In order to assess the ability of different recommendation models when dealing with the cold-start scenario in the task of APC, we use four evaluation metrics, namely R-precision, Normalized Discounted Cumulative Gain (NDCG), Recommended Song Clicks (RSC), and Recall. While we adopt the first three metrics from the RecSys Challenge 2018, we also apply Recall to evaluate model capability in exploring withheld tracks regardless of recommendation order. When expressing the equations of these metrics, we use G and R to denote the ground truth of withheld tracks and the list of recommendations, respectively.

– *R-precision* (Zamani et al. 2019) is the number of relevant tracks among the ground truth divided by the number of withheld tracks. Regardless of their order, the metric rewards the retrieved relevant tracks within the length of the ground truth. R-precision is calculated as follows:

Table 3 Basic statistics of the MPD

Property	Value
Number of playlists	1,000,000
Number of tracks	66,346,428
Number of unique tracks	2,262,292
Number of unique albums	734,684
Number of unique artists	295,860
Number of unique titles	92,944
Number of unique normalized titles	17,381
Number of playlists with descriptions	18,760
Average playlist length	66.35

$$R\text{-}Precision = \frac{\left| G \bigcap R_{1:|G|} \right|}{|G|} \qquad (3)$$

– *NDCG* (Järvelin and Kekäläinen 2002), a metric originally proposed to assess information retrieval systems, has been widely used in MRS to evaluate the ranking of recommended tracks. Discounted Cumulative Gain (DCG) is defined as:

$$DCG = \sum_{i=1}^{|R|} \frac{rel_i}{\log_2(i+1)} \qquad (4)$$

where rel_i is the relevancy label for the track ranked at position i for the playlist. NDCG is the normalization of DCG by the Ideal DCG (IDCG), the scenario in which the recommended tracks are the actual ground truth and also perfectly ranked. Accordingly, NDCG is calculated as follows:

$$NDCG = \frac{DCG}{IDCG} \qquad (5)$$

– *RSC* Zamani et al. (2019) is a measure related with the Recommended Songs feature of the Spotify, which suggests 10 tracks for a target playlist. The users of the Spotify platform can utilize this feature to produce 10 more suggestions. Simply, RSC is the number of refreshes required until a relevant track is encountered. RSC is calculated as follows:

$$RSC = \left\lfloor \frac{\arg\min_i \left\{ R_i : R_i \in G \right\} - 1}{10} \right\rfloor \qquad (6)$$

– *Recall* is the fraction of recommended relevant tracks among all recommendations regardless of any order. Although R-precision and Recall seem quite similar, they differ in the search space of recommendations as the former metric examines the tracks within a sub-list

Fig. 2 Cold-start scenario simulation with a LOOCV-based testing approach
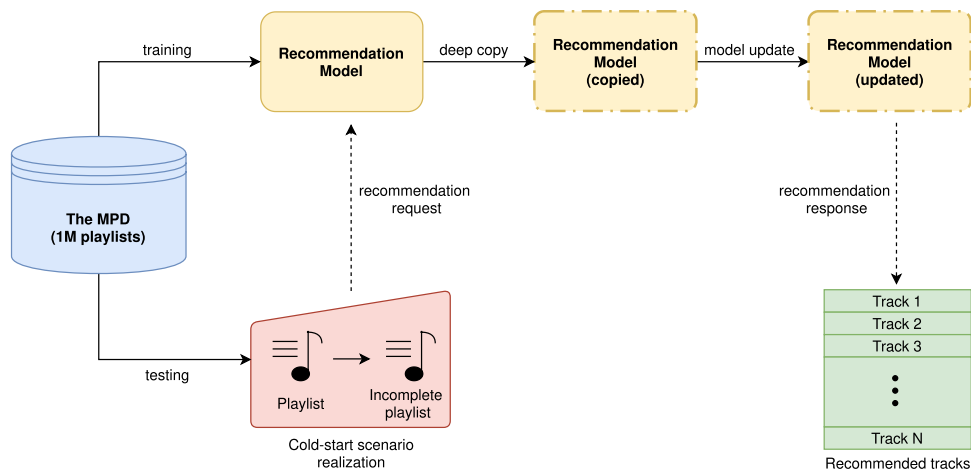
of recommendations limited by the number of withheld tracks. Recall is defined as:

$$Recall = \frac{|G \bigcap R|}{|G|} \tag{7}$$

In terms of recommendation performance and accuracy, the higher R-precision, NDCG, and Recall, the better. On the contrary, the lower RSC, the better (since relevant items should be encountered in earlier suggestion requests made by the target user).

5 Experimental results and analyses

In order to understand the usability and effectiveness of playlist titles to deal with the cold-start problem in APC, we examine the recommendation performance of three different approaches described in Sect. 3. In our experiments, we test every playlist instance in the MPD (i.e., one million playlists) by following the aforementioned LOOCV strategy. As in the RecSys Challenge 2018, we choose the number of recommendations to be made for a playlist as 500.

This section presents our experimental results and analyses in four main perspectives: (i) recommendation performance of the deployed models, (ii) hidden content in the titles of perfectly extended playlists, (iii) characteristics of top-performing title clusters, and (iv) common user preferences when naming music sessions. For the remaining part, *GTP*, *TM*, and *NTM* denote global track popularity model, title matching model, and normalized title matching model, respectively.

5.1 Model-based comparison of overall recommendation performance

The primary step to understand the importance of user-generated titles for cold-start APC is to demonstrate the overall performance of deployed models across the entire dataset. Sampling with each model separately, we first create a recommendation list (with a maximum of 500 tracks) for each playlist in the MPD. Then, we evaluate model accuracy by comparing the actual tracks of the playlists with the recommended tracks for those playlists. Table 4 presents the average scores for the entire dataset on each evaluation metric basis. According to these results, *NTM* model that clusters playlist titles with respect to title similarity provides the best recommendation performance. *TM* model, matching playlist titles directly, also produces comparable recommendations close to the best results. Conversely, *GTP* achieves very low accuracy, which reveals that track popularity should not be used as a standalone recommendation model. Therefore, it can be said that when naming playlists, users prefer common and descriptive expressions, which makes playlist titles a convenient source of information for cold-start APC approaches (Schedl et al. 2018).

Although the title aggregation approach in *NTM* achieves the best recommendation performance in general, there are some cases where the other approaches provide better recommendations for some playlists. In Fig. 3, we present pairwise comparisons of the deployed models, in which blue and

Table 4 Average recommendation performances of deployed models for all playlists in the MPD

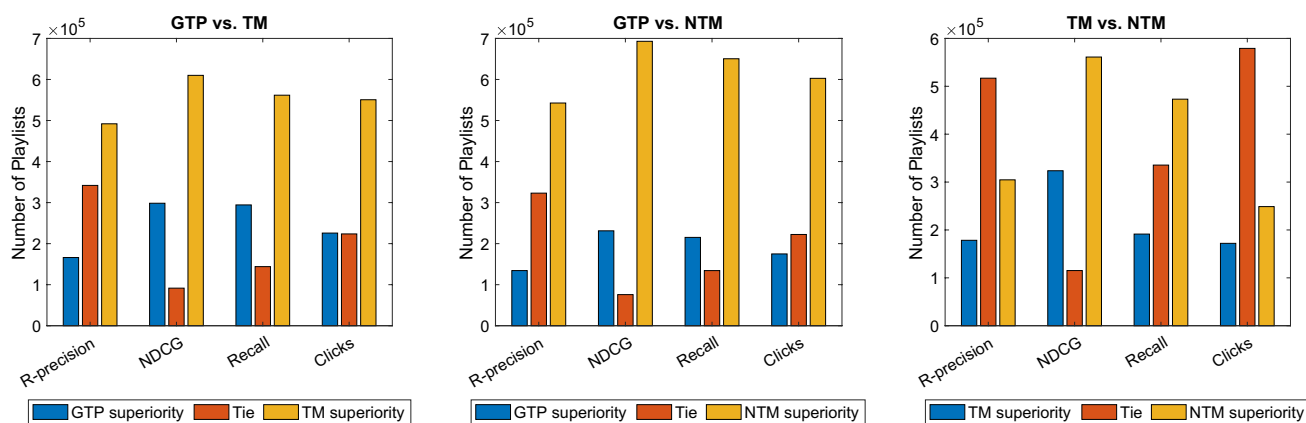| Model	R-precision@500	NDCG@500	Recall@500	RSC@500
GTP	0.027	0.082	0.135	19.338
TM	0.09	0.185	0.252	12.633
NTM	**0.101**	**0.212**	**0.294**	**9.307**

Fig. 3 Pairwise superiority comparison of the deployed recommendation models

Table 5 Recommendation sufficiency of deployed models to provide 500 tracks for each playlist in the MPD

Model	# of average recommendations	Full recommendations[1] # of playlists	Empty recommendations[2] # of playlists
GTP	500	1,000,000 (100%)	0 (0%)
TM	439.04	806,197 (80.61%)	43,459 (4.34%)
NTM	492.53	965,626 (96.56%)	1070 (0.1%)

[1] Playlists that are provided with exactly 500 recommended tracks.

[2] Playlists that cannot be provided with any recommendations

yellow bars show the superiority of compared models over each other, and the red bars indicate ties. When we examine these comparisons in terms of NDCG, we see that *GTP* and *TM* can surpass *NTM* in 23.1% and 32.3% of the entire data set, respectively. As the figure suggests, similar inferences can also be made for the rest of the evaluation metrics. Consequently, the fact that different models can perform better in different scenarios (i.e., playlists) shows us a selective or hybrid approach may be more effective in addressing the cold-start APC problem.

One remarkable aspect of recommendation performance is the ability of a model to produce the required number of recommendations. While *GTP* is an independent model and obviously has the capacity to provide any number of recommended tracks, *TM* and *NTM* may not produce sufficient recommendations in extreme cold-start cases where title-based similarities cannot be calculated. In Table 5, we present to what extent these models meet recommendation requests for all playlists in the MPD.

As illustrated in Table 5, normalizing playlist titles in *NTM* helps to cover the whole title space to a large extent, which enables full recommendation capacity for 96.56% of the playlists in the MPD. In addition, high title coverage also reduces the number of playlists that cannot be provided with any recommendations to a small amount, such as 1070. On the other hand, direct matching of playlist titles in *TM* reduces full recommendation capacity by up to 80.61% and

increases the number of playlists without any recommended tracks to 43,459. As we previously mentioned, while *GTP* has the capacity to make full recommendations for every playlist, the low accuracy in these recommendations tailors *GTP* as a complementary approach rather than a primary recommendation model (Kaya and Bridge 2018; Ferraro et al. 2018).

Title-based cold-start APC approaches experience the inability to generate recommendations for a playlist when the title information cannot be used. In *TM*, this inability indicates that the title of that playlist occurs only once in the MPD. Similarly, *NTM* fails to produce recommendations when a normalized title of a playlist does not match any other title normalizations. Accordingly, the set of 1070 playlists without any recommendations in *NTM* is a subset of 43,459 playlists with empty recommendations in *TM*.

5.2 Title content in perfectly extended cold-start playlists

When a list of recommendations perfectly fits a playlist and gets full credit based on all metrics (i.e., 1 for R-precision, NDCG, and Recall; 0 for RSC), it can be said that the target playlist is *perfectly extended*. In this part, we analyze the reasons behind the complete accuracy in perfectly extended playlists by examining the title content of these instances. Understanding what kind of information is available in the

Table 6 Number of playlist instances that are perfectly extended in the cold-start setting

Model	# of perfectly extended playlists
GTP	0
TM	1613
NTM	1717

titles of these playlists will be a guide for cold-start APC solutions.

As shown in Table 6, 1717 playlists in *NTM* and 1613 playlists in *TM* are perfectly extended. However, *GTP* does not have such a capability. When we inspect whether there are identical playlists in these 3,330 instances, we see that 1516 playlists are common in the result sets of two models. Therefore, there exist 1814 unique perfectly extended playlists in total. In order to understand why the recommended tracks for these playlists have complete accuracy, we examine each playlist title in this group and observe that the corresponding titles point to some specific themes listed below:

– *Artists:* The title of an artist-themed playlist may include the name of related artist directly (e.g., 'ariana grande', 'the notorious b.i.g.', 'one direction').
– *Albums:* Some playlist titles may point to a specific album of an artist. These titles may contain both the artist name and the album name (e.g., 'eminem—curtain call'), as well as only the album name (e.g., '18 months' by Calvin Harris).
– *Compilations:* Some titles may refer to compilation albums or playlists published by various music organizations (e.g., '100 pieces of classical music for your brain', '100 hits of the 80s', 'punk goes pop').

– *Soundtracks:* A title consisting of a movie or TV series name may indicate that the playlist contains music from that production (e.g., 'mamma mia!', 'sons of anarchy', 'la la land').
– *Video games:* If a playlist title includes the name of a video game, the corresponding playlist may likely to contain music from that game (e.g., 'life is strange', 'the last of us').
– *Musicals:* In a musical-themed playlist, the title may contain the name of the corresponding musical (e.g., 'hamilton', 'dear evan hansen').
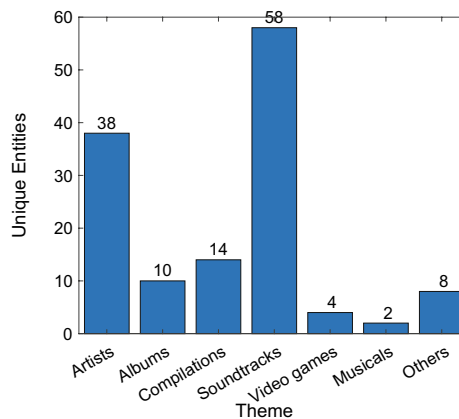
In all perfectly extended playlists, only a few titles (e.g., 'grammy style', 'pirates', 'warriors') cannot be correlated with the themes listed above. Thus, we refer these instances as *Others* for now.

In Fig. 4, we present the specific theme implications in the titles of perfectly extended playlists. A large proportion of these themes (65%, approximately) consist of soundtracks from movies and TV series. This category is followed by artist and compilation themes (11% and 9%, respectively); however, there is quite a difference in the rate of soundtrack themes among perfectly extended playlists. When we inspect the content of these themes, we obtain 134 unique entities in total. In terms of diversity, we observe that soundtrack, artist, and compilation categories include 58, 38, and 14 unique entities, respectively, making them the three most diverse themes in terms of complete recommendations. For the readers interested in the content of themes, we provide a complete list of unique entities in Appendix 1.

In Table 7, we present entity and track statistics of the specific themes in perfectly extended playlists. Considering the length of playlists (i.e., the number of tracks in a playlist, not duration), we see that perfectly extended playlists
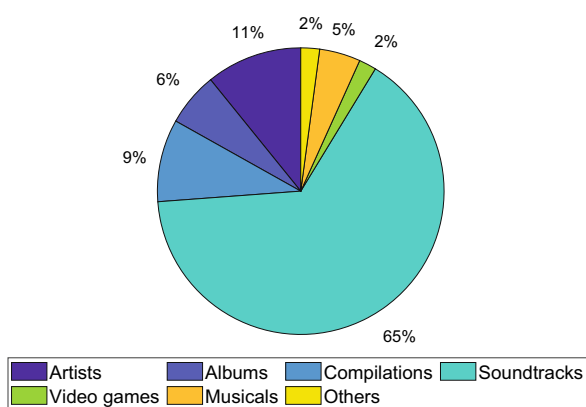
(a) Theme distribution (best viewed in color).

(b) Unique entities in themes.

Fig. 4 Specific themes pointed in the titles of perfectly extended playlists

Table 7 Entity and track statistics of specific themes in perfectly extended playlists

Theme	Playlists covered	Unique entities	Average playlist length	Average unique tracks per entity
Artists	215	38	41.22	54.63
Albums	120	10	28.85	29.4
Compilations	185	14	45.33	44.78
Soundtracks	1293	58	31.11	35.58
Video games	39	4	40.17	51.5
Musicals	92	2	34.19	41.5

are relatively short when compared to the overall dataset average (66.35 tracks, which is shown in Table 3). Also, the track search space is highly local (since the average number of unique tracks per entity ranges from 29.4 to 54.63). For these reasons, it becomes possible to find relevant tracks for playlists pointing to a specific theme.

The analysis of perfectly extended playlists emphasizes the importance of content hidden in playlist titles when dealing with the cold-start APC problem. Although the number of perfectly extended playlists is quite small considering the size of the MPD, it is remarkable that almost all of these playlists can be associated with one of the six specific themes (artists, albums, compilations, soundtracks, video games, and musicals). A title correlated with these themes carries a robust signal to generate candidate recommendations convenient with the user intent. This signal should not be lost during title processing. For example, '100 hits of the 80s' indicates a compilation album; so recommendations for playlists having this title should be first chosen from the corresponding album rather than the popular songs of the 1980s. Thus, a thematic recognition procedure (possibly, named-entity recognition (NER) (Oramas et al. 2017)) should be introduced in music recommender systems during processing and normalizing playlist titles. Such a capability would pave the way for new directions, such as recommending similar artists to an artist-themed playlist or new tracks from other albums of the artist to an album-themed playlist.

5.3 Top-performing title clusters

Depending on the assumption that users name their listening sessions in a similar fashion, *NTM* clusters playlists according to normalized titles and generates recommendations for cold-start instances upon these clusters. When normalizing playlist titles of the MPD, *NTM* employs the standard normalization technique provided by Spotify within the dataset.

As shown in Table 3, this technique results in 17,381 unique normalized titles.

Having 17,381 contextual clusters in total, *NTM* achieves the best recommendation performance among the deployed models. However, which of the clusters created by title normalization are effective in cold-start APC is still unexplored yet. In order to enlighten this question, we calculate R-precision, NDCG, and Recall scores of all clusters by averaging specific measurements of the playlists in each cluster. As illustrated in Fig. 5, a positively skewed distribution can be observed for each metric, which implies that average R-precision, NDCG, and Recall scores are quite low in the vast majority of title clusters. For all three metrics, the number of clusters exceeding the 0.5 threshold is considerably small. However, some title clusters perform quite well compared to the general.

In the rest of this section, we focus on top-performing title clusters in the cold-start APC setting. Initially, we rank all the clusters according to their average NDCG scores, then retrieve *top-n* items in the ranking. We analyze different sizes of clusters (varying *n* from 250 to 1000 in increments of 250, as shown in Table 8) to observe the change of coverage in terms of playlists and unique tracks. As the cluster scope grows, the overall coverage also expands (with less R-precision, NDCG, and Recall scores on average). Consequently, we choose to continue with *top-1000* title clusters for an in-depth analysis.

In addition to the six specific themes described in Sect. 5.2, new identifiers of user intent show up as we delve into more title clusters. The examination of *top-1000* items reveals the following additional content:

- *Festivals:* The title of a playlist may indicate a specific music festival (e.g., 'bonnaroo 2017', 'coachella', 'outside lands 2015'). The music festivals are often named with the year in which they are held.
- *Genres:* A title consisting of a music genre or a dance style is a strong indicator of user preference for playlists (e.g., 'latin trap', 'country 2015', '90s alternative'). It is also common in genre-themed playlists to include year and decade references in the titles.
- *Record labels:* Some playlist titles may refer to a company in music industry, which works in the publishing of music recordings (e.g., 'two steps from hell', 'motown', 'ovoxo'). Such a playlist usually contains tracks from a group of artists who have worked with that company.
- *Time references:* A title might include a time reference (pointing to an earlier time in all *top-1000* clusters) in terms of decades and years. These indicators are usually combined with another theme such as a genre or a festival (e.g., 'reggaeton 2017', '2000s r&b')
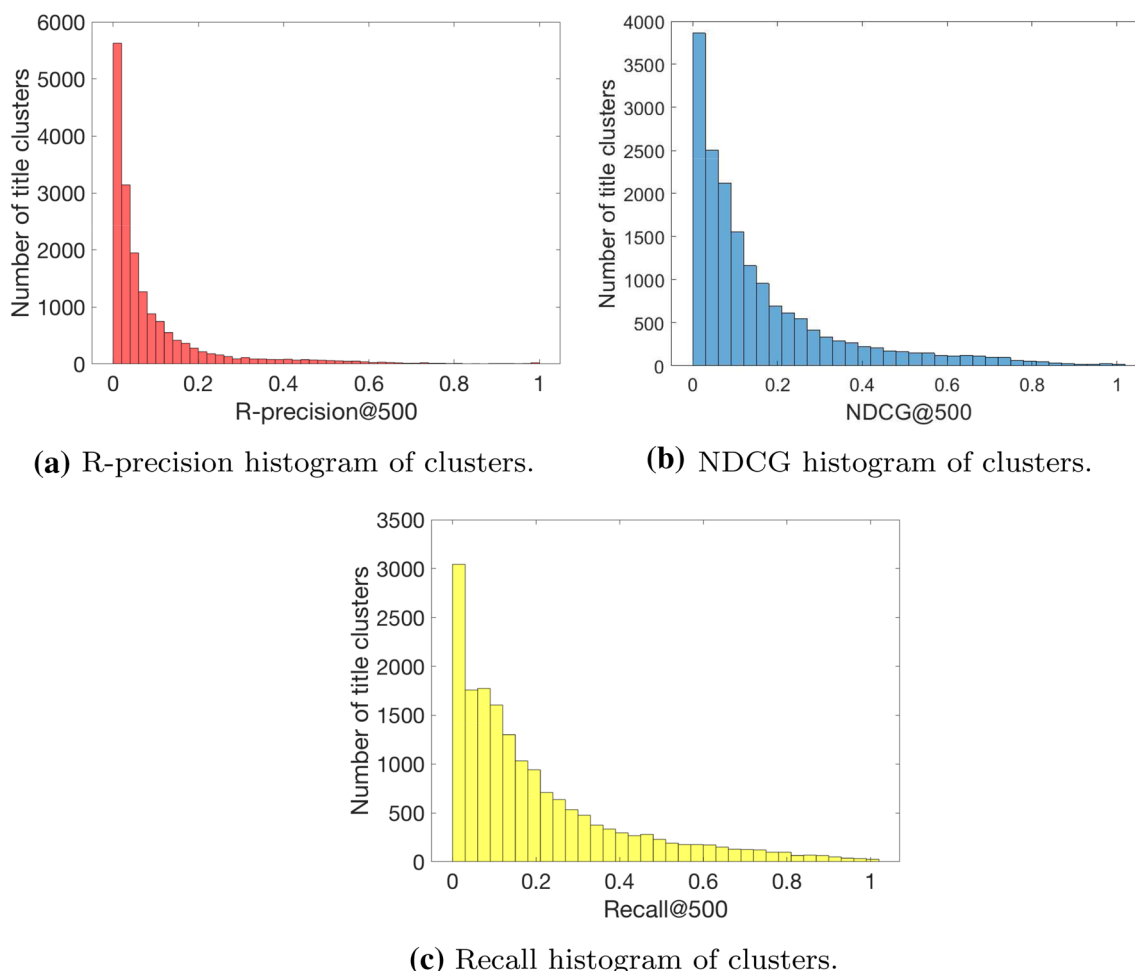
(a) R-precision histogram of clusters.

(b) NDCG histogram of clusters.

(c) Recall histogram of clusters.

Fig. 5 R-precision, NDCG, and Recall score distributions of normalized title clusters

	Cluster scope	Playlists covered	Unique tracks covered	R-precision@500		NDCG@500		Recall@500	
				Min.	Avg.	Min.	Avg.	Min.	Avg.
Table 8 Playlist and track coverage of top-performing title clusters	top-250	8842	44,693	0.494	0.756	0.779	0.873	0.679	0.899
	top-500	19,758	99,029	0.358	0.65	0.691	0.801	0.585	0.84
	top-750	27,800	149,312	0.291	0.59	0.628	0.753	0.495	0.797
	top-1000	34,229	201,987	0.172	0.544	0.563	0.714	0.452	0.762

In the prior analysis considering only perfectly extended playlists, the soundtrack category is the dominant one among the identified themes. Once we relax the scale of success and increase the scope by examining *top-1000* title clusters, the artist category comes to the front. As presented in Fig. 6a, 46% of the successfully extended cold-start playlists are related with an artist. Soundtracks follow this category with a rate of 30%. Among the remaining categories, genres and time references (6% and 5%, respectively) are also notable in terms of the number of playlists covered.

Diversity is also a critical aspect to take into consideration when identifying the themes convenient for cold-start APC approaches utilizing playlist titles. In *top-1000* title clusters, we detect 786 unique entities whose categorical distributions are presented in Fig. 6b. Covering a large proportion of the analyzed playlist, the artist theme also ranks first in terms of diversity with 510 unique entities. The second and third most diverse themes are soundtracks and albums (associated with 93 and 89 unique entities, respectively). In Fig. 6c, we also provide a visualization of the entities with respect to their identified themes.

Fig. 6 Specific themes pointed in top-performing clusters

(a) Theme distribution (best viewed in color).

(b) Unique entities in themes.
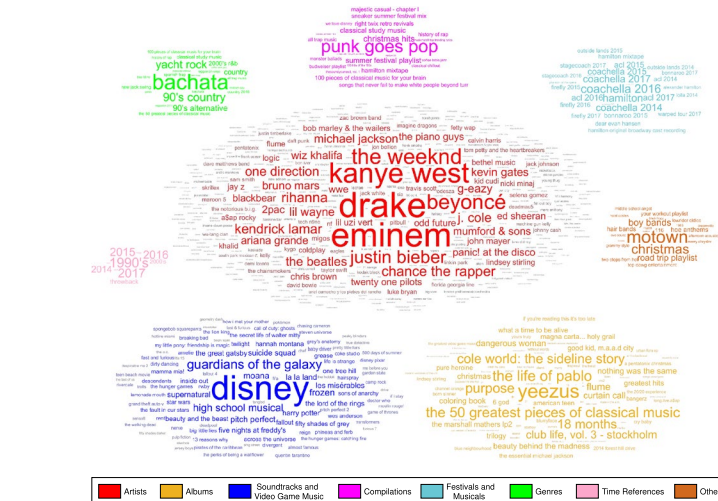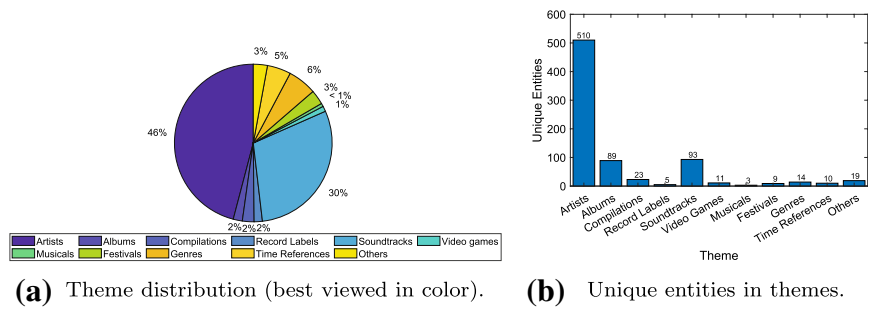
(c) A visualization of the explored title space (best viewed in color).
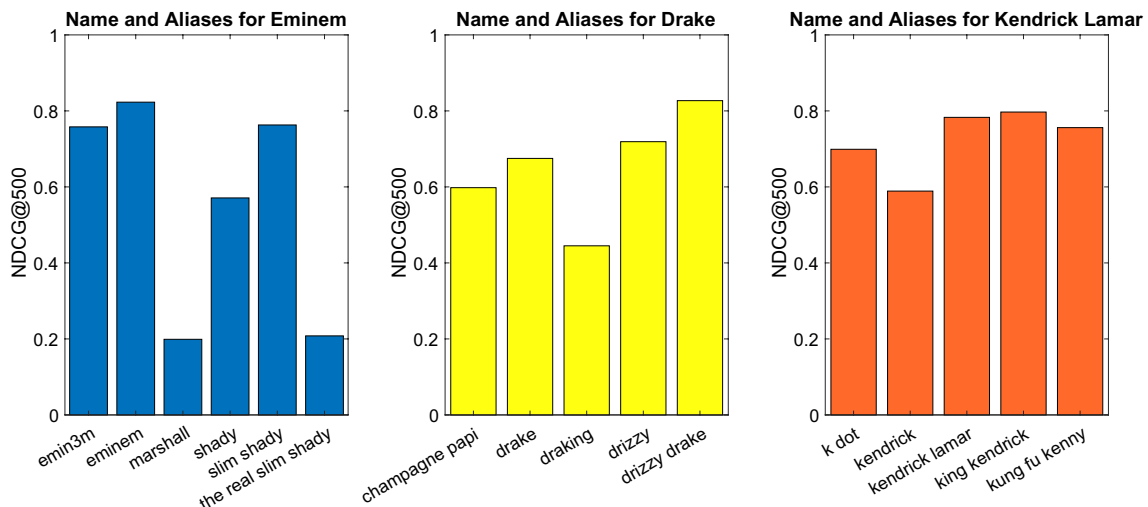

Fig. 7 Three cases of using aliases as an alternative to artist names in the titles of artist-themed playlists

Another thing to note about artist-themed playlists is the frequent usage of aliases (including stage names, nicknames, and abbreviations) in titles. Users are likely to prefer aliases instead of directly using the name of an artist when creating playlists that contain tracks mostly belonging to a single artist. For example, 'emin3m', 'marshall', 'shady', 'slim shady',

Table 9 Some examples of title clusters that point out artists with abbreviations, stage names, or nicknames

Artist	Correlated clusters
Beyoncé	'beyoncé, 'yoncé', 'bey', 'queen b', 'queen bey'.
Drake	'drake', 'champagne papi', 'draking', 'drizzy', 'drizzy drake'.
Eminem	'eminem', 'emin3m', 'marshall', 'shady', 'slim shady', 'the real slim shady'.
Five Finger Death Punch	'five finger death punch', 'ffdp', '5fdp'.
Justin Bieber	'justin bieber', 'bieber fever', 'biebs', 'bieber', 'jb'.
Kendrick Lamar	'kendrick lamar', 'kendrick', 'k dot', 'king kenny', 'kung fu kenny'.
Michael Jackson	'michael jackson', 'king of pop', 'mj'.
Panic! At The Disco	'panic! at the disco', 'panic at the disco', 'panic!', 'panic', 'p!atd', 'patd'.
Taylor Swift	'taylor swift', 't-swizzle', 'tswizzle', 'tswift'.
Twenty One Pilots	'twenty one pilots', '21 pilots', 'twentyonepilots', 'tøp'.

and 'the real slim shady' are all similar clusters that point playlists related to Eminem, a famous American rapper[4]. In Fig. 7, we present three examples of using aliases instead of artist names in playlist titles. When the NDCG scores of artist name and alias clusters are compared, it is seen that aliases are as effective as artist names in understanding user intent and playlist characteristics.

The abovementioned typical user behavior re-emphasizes the need for musical NER in cold-start solutions, which is previously discussed in Sect. 5.2. Similar to artist names, aliases also carry strong signals indicating playlist characteristics. Recognizing such content will help to produce accurate recommendations for cold-start playlists. Some other examples expressing artists with aliases are presented in Table 9.

5.4 Most common title clusters

As aforementioned earlier, the tendency of users to prefer similar terms when naming music playlists allows one million titles in the MPD to be aggregated into 17,318 textual clusters. In general, users prefer some titles (e.g., 'chill', 'country', 'rap') much more than others, which results in a non-uniform playlist distribution among the title clusters. However, frequent preference of a playlist title does not always show that it is a good information source to recommend music tracks; the title may be naturally not valuable for recommender systems, or it may indicate a context where music consumption differs widely (Pichl et al. 2015). For this reason, it would be appropriate to investigate the effectiveness of the most common title clusters in cold-start APC.

In order to explore which terms users prefer most as their playlist titles, we examine the clusters in terms of their playlist coverage. As presented in Table 10, 113 title clusters (each including at least 1000 playlists) cover 28.65% of the

MPD, and the remaining 17,268 clusters cover the rest of the dataset. In the rest of this section, we accept the former group as the most common title clusters and present their recommendation performance in cold-start APC.

When we examine the title preferences of users in the MPD, we see that the most common title clusters comprise of genres, time references, activities, mood and emotions, and some generic terms. In Fig. 8, we present the overall recommendation success of each category in terms of NDCG. Recommendations produced based on common title clusters that indicate a music genre or a certain time period have reasonable accuracy for the cold-start problem in APC. Interestingly, clusters pointing to past, such as 'throwbacks', '90s', and 'oldies', perform better than more recent content like 'new' and 'summer 2017'. On the other hand, the clusters that can be associated with daily activities and emotions of users (e.g., 'driving', 'running', 'work', 'workout', 'happy', 'sad', and 'chill') do not imply a common sense, which makes them limited in title-based cold-start approaches. Furthermore, terms such 'jams', 'my music', 'vibes', 'mix', and 'good songs', which are frequently preferred by users when naming playlists, do not contain any useful information and are not much valuable for recommender systems. For the interested readers, details about the most common clusters are included in Appendix 2.

Table 10 Comparison of clusters according to playlist coverage

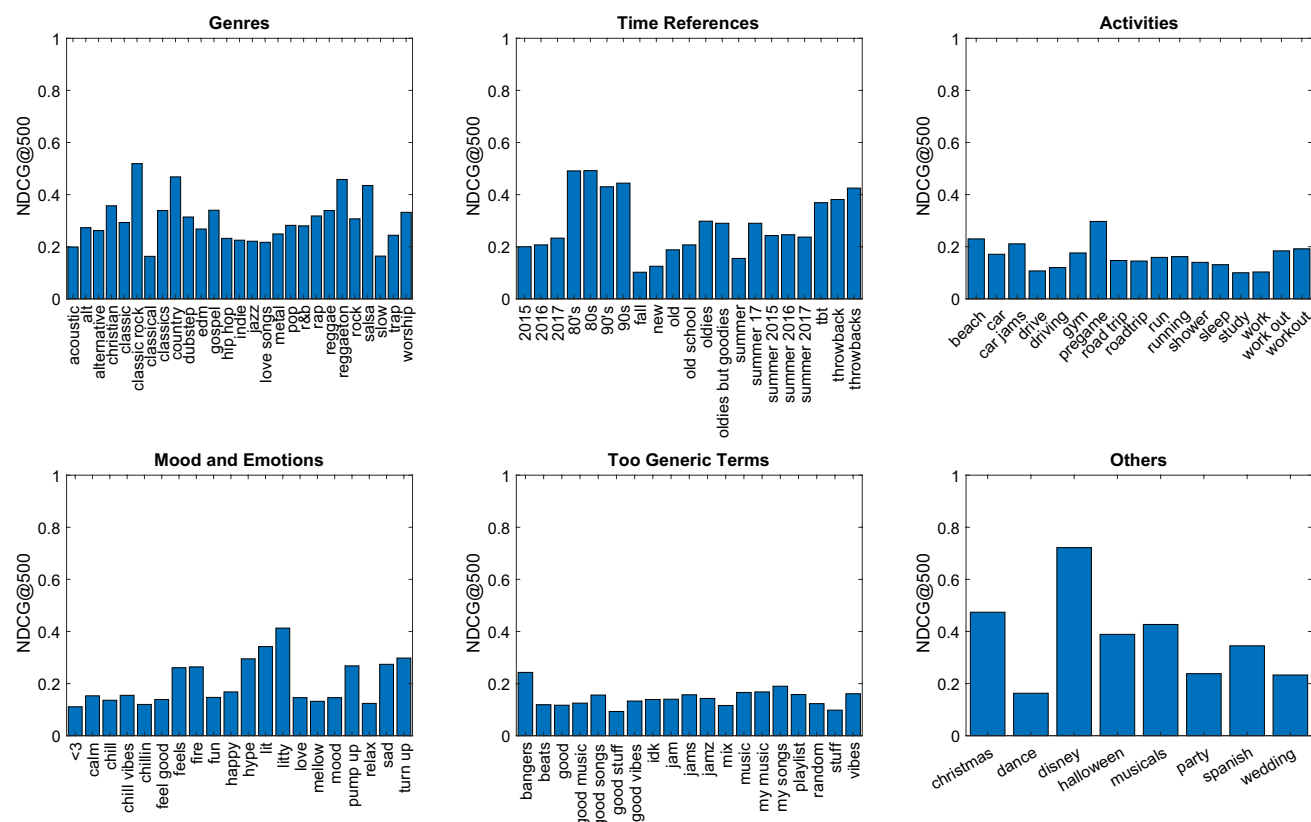|	Total clusters	Playlists covered	MPD coverage (%)
Clusters with at least 1000 playlists	113	286,538	≈ 28.65
Clusters with less than 1000 playlists	17,268	713,462	≈ 71.35

---

[4] https://en.wikipedia.org/wiki/Eminem.

Fig. 8 Recommendation performance of the most common clusters in terms of NDCG

6 Discussion and special notes

A playlist title can help to alleviate the cold-start problem encountered in new playlists by giving hints about the intended purpose of users (Zamani et al. 2019). Experimentally, we illustrate that aggregating titles into textual clusters is a promising approach to make use of the common attitude of users in naming their playlists. In this section, we summarize and discuss our findings and provide special notes on how to model a title-aware approach to the problem.

Aim and scope This study investigates the whens and hows of title usability in recommending music to newly created playlists without any tracks. Following a testing strategy based on LOOCV (described in Sect. 4.2), we simulate the cold-start scenario for one million playlists and evaluate the accuracy of different recommendation approaches on each of these instances. Considering the magnitude of the test set, it can be said that the results and analysis obtained are comprehensive and reveal the general attitude of users in their title preferences.

Analyses and findings Users have similar preferences when naming their music listening sessions. The experimental results of the study show that title clusters built upon this behavioral similarity constitute a reasonably alternative solution for the cold-start APC problem.

Especially in cases where titles indicate specific themes such as artists, albums, compilations, and soundtracks, title aggregation can result in highly accurate recommendations. Therefore, recognition of such content is of special importance for further recommendation directions and improvements. When playlist coverage of title clusters is examined, it is seen that playlists are non-uniformly distributed among clusters; users prefer a small group of titles much more than a wide variety of others. Furthermore, the playlist titles that are very frequently preferred by users can also provide promising results, as long as they contain information implying a collective sense in music. Otherwise, they become a bit useless for recommendation tasks in MRS.

Limitations From a critical point of view, we can discuss two limitations of the study: (i) the simplicity of the employed title normalization technique, and (ii) the semi-automated title identification process. Firstly, when creating textual clusters from playlist titles, we employ Spotify's normalization method provided within the MPD to present reproducible results. However, the method is not optimal and needs fair improvements. For example, it cannot detect that '80s' and 'eighties' point to the same theme. Thus, better textual clustering can be achieved by using sophisticated NLP techniques (Zamani et al. 2019).
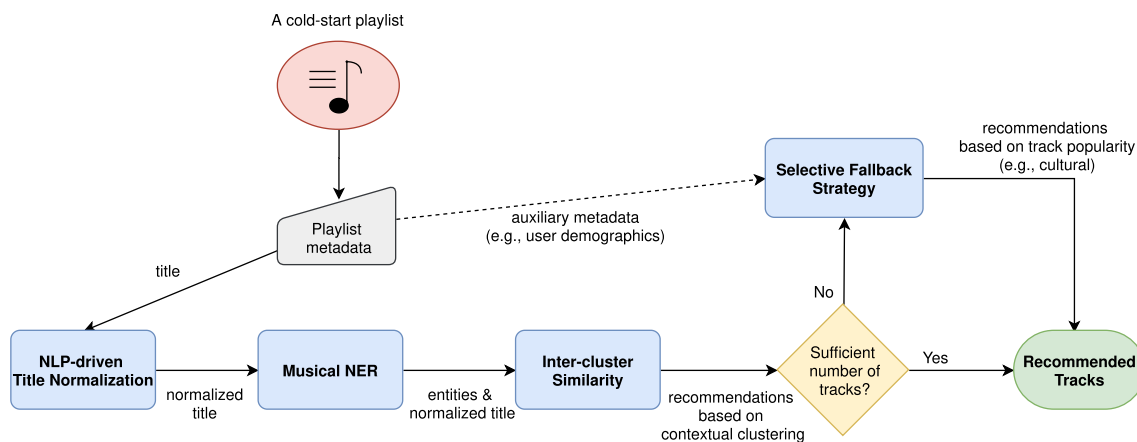
Fig. 9 The flowchart of an effective title-aware recommendation approach to the cold-start APC problem

Although an advanced normalization is likely to increase the recommendation accuracy of *NTM*, it will not have much effect on overall analyses and findings. Secondly, the identification of content hidden in titles is carried out semi-automatically during the study. While artists and albums can be automatically detected by querying the MPD, it is not possible to identify many other themes, such as soundtracks or video game music, without using an external knowledge base. Therefore, some titles are manually annotated, which is a quite time-consuming task that also prevents the categorization of all titles.

In line with the abovementioned findings and discussions, the following notes can be used as a guideline to develop an effective title-aware recommendation approach to the cold-start APC problem:

1. A preprocessing step (i.e., title normalization) is essential to create textual clusters from playlist titles. Since better clustering implies better recommendation accuracy, the preprocessing capabilities can be extended by utilizing NLP techniques. It is noteworthy that this step is prone to information loss, as some special characters (e.g., emojis and symbols) may carry potential signals about user intent.

2. Revealing the content hidden in playlist titles is critical to provide coherent recommendations for cold-start playlists. A recognition procedure, such as NER, should be introduced to detect entities in music and related fields like movies and TV shows.

3. In addition to effective title normalization and musical NER mechanisms, inter-cluster relationships should also be considered. Clusters that are syntactically or semantically similar to each other (e.g., 'spanish rock' and 'rock en espanol', 'gym' and 'workout', 'summa 2k17' and 'summer 17') may carry a common sense for cold-start recommendations.

4. A fallback strategy should be formed for cases where no inference can be made from playlist titles. Besides recommending popular music, this secondary approach may reflect geographic or cultural preferences (Schedl and Schnitzer 2013; Hong et al. 2019) by incorporating user demographics. In addition, this strategy can be selectively used for playlists with titles like 'good stuff', 'my songs', and 'random', which are experimentally proven to be invaluable for recommender systems.

In Fig. 9, we present the flowchart of a title-aware recommendation approach to the cold-start APC problem, based on the proposed guidelines. Given a cold-start playlist, the primary motivation is to maximize the inference attained from user-generated content, which is a key factor in improving recommendation accuracy in MRS. As discussed earlier in Sect. 2.1, the existing studies attempt to uncover latent relationships between title similarities and track co-occurrences by various methods, including matrix factorization, neural networks, and nearest neighbor search (Zamani et al. 2019). Regardless of the applied techniques, these approaches do not take into account the semantic content of playlist titles; they solely focus on vector space representation of title terms. Therefore, semantic information, which is quite meaningful for recommender systems, cannot be effectively utilized. The proposed guidelines can also be followed to introduce title-awareness into recommendation processes of the existing methodologies and overcome the specified limitation.

7 Conclusion

The cold-start problem, caused by the lack of predictive data, makes the task of APC more challenging than usual. One possible approach to alleviate the problem is to

utilize playlist titles and incorporate user-generated data into the recommendation process. However, these titles may not always be informative about the intended purpose or characteristics of music playlists. Specifically, users tend to prefer short, common, and generic descriptions when naming their music listening sessions. For this reason, it should be revealed in which situations title-based approaches can produce accurate recommendations for playlists suffering from the cold-start.

In this study, we investigate the effectiveness of playlist titles as an additional information source to deal with cold-start APC. Employing three naive recommendation models, we conduct experiments on one million music playlists from the Spotify platform. Based on empirical results, we show that aggregating titles into contextual clusters is a moderate strategy to alleviate the cold-start problem in APC regarding both recommendation accuracy and sufficiency. Then, examining playlist instances with accurate recommendations, we clarify the underlying reasons why certain titles are well-indicators of user intent. Our experimental results exhibit that titles pointing a specific theme can successfully narrow down the search space of recommendations, which makes these titles practical for cold-start APC solutions. Among various concepts such as soundtracks, albums, and genres, artists theme becomes prominent in terms of both musical diversity and user preference. In addition to assessing the recommendation capabilities of playlist titles in cold-start APC, we also examine users' common attitudes in naming their music sessions. Within the wide result set obtained by clustering titles according to their spellings, we show that a considerably small group (including genres, time references, activities, mood, emotions, and some generic terms) can reflect the majority of user preferences. Nonetheless, the correlation between the common preference of a title and its importance in recommendation accuracy is quite weak.

We believe that the current work provides a proof of concept for the utilization of playlist titles in cold-start APC, and our findings can be a guideline to MRS research and music industry. The natural next step is to develop title-aware recommendation models that can compete with the state-of-the-art techniques for the cold-start problem in APC.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Appendix 1: Entities Identified in Perfectly Extended Playlists

See Table 11.

Appendix 2: The most common title clusters

See Table 12.

Table 11 Unique entities and corresponding themes of perfectly extended playlists

Theme	Entities		
Artists	Antonio Vivaldi	Ariana Grande	Black Label Society
	Bonobo	Calvin Harris	Camila Cabello
	Daft Punk	Demi Lovato	Eminem
	Flume	Gnash	Halsey
	Kanye West	Kevin Gates	Khalid
	La Dispute	London Philharmonic Orchestra	Luan Santana
	Lynyrd Skynyrd	Madeon	Michael Jackson
	Mumford & Sons	NCT	NF
	Nick Drake	NSYNC	Odd Future
	One Direction	RAC	Rod Stewart
	Sam Smith	Shane & Shane	The Civil Wars
	The Naked and Famous	The Notorious B.I.G.	Tiësto
	Yann Tiersen	Zero 7	
Albums	18 Months (by Calvin Harris)	Club Life, Vol. 3—Stockholm (by Tiësto)	Curtain Call (by Eminem)
	Dangerous Woman (by Ariana Grande)	Flume (by Flume)	Greatest Hits (by The Notorious B.I.G.)
	The Life of Pablo (by Kanye West)		
	The 50 Greatest Pieces of Classical Music (by London Philharmonic Orchestra)		
	The Essential Michael Jackson (by Michael Jackson)		
	The Greatest Video Game Music (by London Philharmonic Orchestra)		
Compilations	100 Hits of the 80s	100 Pieces of Classical Music for Your Brain	All Trap Music
	Christmas Hits	Classical Chillout	Classical Study Music
	Hamilton Mixtape	José José, Un Tributo 1 & 2	Majestic Casual—Chapter I
	Punk Goes Pop	Right TWIX Retro Revivals	TheSoundYouNeed, Vol. 1
	Vintage Café: Lounge & Jazz Blends	We Love Disney	
Soundtracks	500 Days of Summer	About Time	Across the Universe
	Amélie	Beauty and the Beast	Begin Again
	Camp Rock	Chef	Deadpool
	Descendants	Despicable Me 2	Dirty Dancing
	Divergent	Endless Love	Fifty Shades Darker
	Fifty Shades of Grey	Frozen	Furious 7
	Garden State	Grease	Guardians of the Galaxy
	Hairspray	Hannah Montana	Harry Potter
	High School Musical	If I Stay	Into the Woods
	Juno	La La Land	Lemonade Mouth
	Les Misérables	Mamma Mia!	Moana
	Moulin Rouge!	Nerve	Oliver!
	Phineas and Ferb	Pirates of the Caribbean	Pitch Perfect
	Pitch Perfect 2	Pulp Fiction	Rent
	RWBY	Sing Street	Sons of Anarchy
	Star Wars	Steven Universe	Suicide Squad
	Suits	Teen Beach Movie	The Fault in Our Stars
	The Great Gatsby	The Hobbit	The Hunger Games: Catching Fire
	The Lion King	The Secret Life of Walter Mitty	Trolls
	Tron		
Video games	Call of Duty: Ghosts	Life is Strange	Mass Effect
	The Last of Us		
Musicals	Dear Evan Hansen	Hamilton	

Table 12 Recommendation performance and coverage of most common title clusters

Cluster name	NDCG@500	Playlists covered	Cluster name	NDCG@500	Playlists covered
chill	0.136	10,000	2016	0.207	1985
country	0.468	10,000	sad	0.274	1980
rap	0.318	8493	study	0.1	1962
workout	0.192	8481	r&b	0.28	1934
oldies	0.298	8146	pregame	0.297	1921
christmas	0.474	8015	driving	0.12	1896
rock	0.307	6848	jamz	0.143	1884
party	0.238	6157	mix	0.116	1860
throwback	0.381	5883	good vibes	0.133	1826
jams	0.157	5063	my music	0.168	1826
worship	0.332	5052	roadtrip	0.145	1786
summer	0.155	4907	80s	0.492	1756
feels	0.261	4677	90s	0.444	1732
new	0.125	4612	mellow	0.132	1709
disney	0.722	4186	< 3	0.111	1683
lit	0.342	4124	mood	0.146	1662
throwbacks	0.425	4030	tbt	0.369	1635
music	0.166	3886	jazz	0.221	1633
sleep	0.131	3513	work out	0.184	1633
vibes	0.161	3500	drive	0.107	1626
running	0.162	3465	chillin	0.12	1620
edm	0.268	3322	80's	0.491	1595
classic rock	0.519	3260	classic	0.293	1532
classics	0.339	3207	indie	0.225	1532
gym	0.176	3144	trap	0.244	1514
dance	0.163	3139	idk	0.139	1487
road trip	0.147	3022	halloween	0.389	1466
pop	0.282	2879	summer 2016	0.246	1466
run	0.159	2875	beach	0.23	1438
car	0.171	2841	fun	0.147	1361
hype	0.295	2767	90's	0.43	1348
old school	0.207	2720	calm	0.153	1340
work	0.103	2659	turn up	0.298	1316
random	0.123	2608	pump up	0.268	1290
wedding	0.233	2585	summer 17	0.29	1266
2017	0.233	2551	fall	0.102	1260
hip hop	0.232	2478	2015	0.2	1247
love	0.146	2467	acoustic	0.199	1245
shower	0.14	2280	love songs	0.217	1237
alternative	0.262	2249	gospel	0.34	1236
christian	0.357	2238	stuff	0.098	1203
reggae	0.339	2237	fire	0.264	1200
relax	0.124	2209	good stuff	0.093	1171
spanish	0.345	2167	old	0.188	1167
happy	0.168	2134	dubstep	0.314	1159
feel good	0.139	2120	slow	0.164	1158
classical	0.163	2036	alt	0.273	1155
summer 2017	0.237	2030	jam	0.14	1133
playlist	0.158	1998	musicals	0.427	1129
oldies but goodies	0.29	1996	chill vibes	0.155	1128

References

Antenucci S, Boglio S, Chioso E, Dervishaj E, Kang S, Scarlatti T, Dacrema MF (2018) Artist-driven layering and user's behaviour impact on recommendations in a playlist continuation scenario. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, BC, Canada, pp 4:1–4:6

Bobadilla J, Ortega F, Hernando A, Bernal J (2012) A collaborative filtering approach to mitigate the new user cold start problem. Knowl-Based Syst 26:225–238. https://doi.org/10.1016/j.knosys.2011.07.021

Bonnin G, Jannach D (2014) Automated generation of music playlists: survey and experiments. ACM Comput Surv 47(2):26:1–26:35. https://doi.org/10.1145/2652481

Chen CW, Lamere P, Schedl M, Zamani H (2018) Recsys challenge 2018: automatic music playlist continuation. In: Proceedings of the 12th ACM conference on recommender systems, Vancouver, British Columbia, Canada, pp 527–528

Chung CH, Chen Y, Chen HH (2017) Exploiting playlists for representation of songs and words for text-based music retrieval. In: ISMIR, pp 478–485

Clark E, Araki K (2011) Text normalization in social media: progress, problems and applications for a pre-processing system of casual english. Proc Soc Behav Sci 27:2–11. https://doi.org/10.1016/j.sbspro.2011.10.577

Elahi M, Ricci F, Rubens N (2016) A survey of active learning in collaborative filtering recommender systems. Comput Sci Rev 20(C):29–50. https://doi.org/10.1016/j.cosrev.2016.05.002

Faggioli G, Polato M, Aiolli F (2018) Efficient similarity based methods for the playlist continuation task. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, BC, Canada, pp 15:1–15:6

Ferraro A, Bogdanov D, Yoon J, Kim K, Serra X (2018) Automatic playlist continuation using a hybrid recommender system combining features from text and audio. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, BC, Canada, pp 1–5

Gillhofer M, Schedl M (2015) Iron maiden while jogging, debussy for dinner? In: MultiMedia modeling, pp 380–391

Hong M, An S, Akerkar R, Camacho D, Jung JJ (2019) Cross-cultural contextualisation for recommender systems. J Ambient Intell Humaniz Comput. https://doi.org/10.1007/s12652-019-01479-9

Järvelin K, Kekäläinen J (2002) Cumulated gain-based evaluation of IR techniques. ACM Trans Inf Syst 20(4):422–446. https://doi.org/10.1145/582415.582418

Kallumadi S, Mitra B, Iofciu T (2018) A line in the sand: recommendation or ad-hoc retrieval? In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, Canada, pp 7:1–7:6

Kaya M, Bridge D (2018) Automatic playlist continuation using sub-profile-aware diversification. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, BC, Canada, pp 1:1–1:6

Kelen DM, Berecz D, Béres F, Benczúr AA (2018) Efficient k-nn for playlist continuation. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, BC, Canada

Kim J, Won M, Liem CCS, Hanjalic A (2018) Towards seed-free music playlist generation: enhancing collaborative filtering with playlist title information. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, BC, Canada

Koren Y, Bell R, Volinsky C (2009) Matrix factorization techniques for recommender systems. Computer 42(8):30–37. https://doi.org/10.1109/MC.2009.263

Lee J (2011) How similar is too similar?: exploring users' perceptions of similarity in playlist evaluation. In: Proceedings of the 12th international society for music information retrieval conference, ISMIR 2011, pp 109–114

Ludewig M, Kamehkhosh I, Landia N, Jannach D (2018) Effective nearest-neighbor music recommendations. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, BC, Canada

Oramas S, Ferraro A, Correya AA, Serra X (2017) Mel: a music entity linking system. In: Hu X, Cunningham SJ, Turnbull D, Duan Z. ISMIR 2017. 18th International society for music information retrieval conference; 2017 Oct 23-27; Suzhou, China. [Canada]: ISMIR; 2017.[2] p

Park YJ, Tuzhilin A (2008) The long tail of recommender systems and how to leverage it. In: Proceedings of the 2008 ACM conference on recommender systems, Lausanne, Switzerland, pp 11–18

Pichl M, Zangerle E, Specht G (2015) Towards a context-aware music recommendation approach: what is hidden in the playlist name? In: 2015 IEEE international conference on data mining workshop, Atlantic City, USA, pp 1360–1365

Pichl M, Zangerle E, Specht G (2017) Improving context-aware music recommender systems: beyond the pre-filtering approach. In: Proceedings of the 2017 ACM on international conference on multimedia retrieval, Bucharest, Romania, pp 201–208

Rashid AM, Karypis G, Riedl J (2008) Learning preferences of new users in recommender systems: an information theoretic approach. SIGKDD Explor Newsl 10(2):90–100. https://doi.org/10.1145/1540276.1540302

Rubtsov V, Kamenshchikov M, Valyaev I, Leksin V, Ignatov DI (2018) A hybrid two-stage recommender system for automatic playlist continuation. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, Canada, pp 16:1–16:4

Schedl M, Schnitzer D (2013) Hybrid retrieval approaches to geospatial music recommendation. In: Proceedings of the 36th International ACM SIGIR conference on research and development in information retrieval, Dublin, Ireland, pp 793–796

Schedl M, Flexer A, Urbano J (2013) The neglected user in music information retrieval research. J Intell Inf Syst 41:523–539. https://doi.org/10.1007/s10844-013-0247-6

Schedl M, Knees P, McFee B, Bogdanov D, Kaminskas M (2015) Music recommender systems. In: Ricci F, Rokach L, Shapira B (eds) Recommender systems handbook. Springer, Boston, pp 453–492

Schedl M, Zamani H, Chen CW, Deldjoo Y, Elahi M (2018) Current challenges and visions in music recommender systems research. Int J Multimed Inf Retr 7:95–116. https://doi.org/10.1007/s13735-018-0154-2

Vall A, Dorfer M, Eghbal-zadeh H, Schedl M, Burjorjee K, Widmer G (2019) Feature-combination hybrid recommender systems for automated music playlist continuation. User Model User-Adapted Interaction 29(2):527–572. https://doi.org/10.1007/s11257-018-9215-8

Volkovs M, Rai H, Cheng Z, Wu G, Lu Y, Sanner S (2018) Two-stage model for automatic playlist continuation at scale. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, Canada, pp 9:1–9:6

Yang H, Jeong Y, Choi M, Lee J (2018) Mmcf: Multimodal collaborative filtering for automatic playlist continuation. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, Canada, pp 11:1–11:6

Zamani H, Schedl M, Lamere P, Chen CW (2019) An analysis of approaches taken in the ACM recsys challenge 2018 for automatic music playlist continuation. ACM Trans Intell Syst Technol 10:57:1–57:21. https://doi.org/10.1145/3344257

Zangerle E, Pichl M, Gassler W, Specht G (2014) #nowplaying music dataset: Extracting listening behavior from twitter. In: Proceedings of the first international workshop on internet-scale multimedia management, Orlando, Florida, USA, pp 21–26

Zhao X, Song Q, Caverlee J, Hu X (2018) Trailmix: An ensemble recommender system for playlist curation and continuation. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, BC, Canada, pp 8:1–8:6

Zhu L, He B, Ji M, Ju C, Chen Y (2018) Automatic music playlist continuation via neighbor-based collaborative filtering and discriminative reweighting/reranking. In: Proceedings of the ACM recommender systems challenge 2018, Vancouver, Canada, pp 10:1–10:6