Combining Objective Measures of Playlist Quality to Evaluate Automatically Generated Playlists

Jennifer Logan

4th Year Project Report
Computer Science
School of Informatics
University of Edinburgh

2021

Abstract

As digital music consumption increases, so does the importance of well-performing music recommender systems. A particular area of this, Automatic Playlist Generation, uses collections of songs known as playlists as a recommendation item. Music, generally, as a recommendation item is extremely subjective and user enjoyment depends on their music tastes, the feelings certain music evokes in them, and their mood. As a result, the success of music recommender systems is hard to evaluate. This is particularly true for playlist generation, as the content a user will want the playlist to include is entirely dependent on their purpose in creating it. Current evaluation techniques are either human based but very limited, or use rigid information retrieval metrics to evaluate approaches to playlist generation based on pre-existing datasets. This project proposes a new evaluation metric that aims to optimise playlist quality based on user preference, bridging the gap between human and machine evaluation. An overview of the field and its current evaluation techniques and metrics is given, and then the new metric is proposed and discussed. The metric achieves promising results on a small user study, and indicates the metric is a good starting point for further work in the creation of a playlist quality measure.

Acknowledgements

Firstly, thank you to my supervisor Pavlos Andreadis for his invaluable guidance and advice in the undertaking of this project. It would be far worse without his expertise.

Thank you to my wonderful family for their unwavering support and encouragement, especially over the past four years. Without them I would never have made it this far through my degree and I can't imagine it's been very fun listening to me complain the whole time.

Finally, thank you to my incredible friends Kim and George for their advice, friendship, company, and willingness to always help me with my various academic endeavours.

Table of Contents

Chapter 1

Introduction

Due to the popularity of digital streaming platforms as a means of music consumption, and particularly the use and creation of playlists, an area in music recommender systems research of Automatic Playlist Generation and a sub-problem Automatic Playlist Continuation, has emerged. Multiple studies [13, 18, 14] found playlists were a main way that users consumed music, and platforms such as Spotify continually make personalised playlists for its users [16]. Automatic Playlist Generation (APG) has a main application for these streaming platforms as an automated approach to this. APG takes some starting information about a song or a user and creates (generates) a playlist of a certain length to match some criteria. Automatic Playlist Continuation (APC) has a starting point of an already completed or partially completed playlist, and its goal is to suggest songs for the user to continue their playlist with.

A key issue in both of these areas is evaluation. Human evaluation is best for music recommendation problems due to the differing ways in which machines and humans describe music (the 'Semantic Gap' [27]). However, due to the current scale of the music industry, and streaming platforms, a user study large enough to evaluate the required volume of data would be extremely costly and time consuming. There is no standard procedure or measure for evaluating approaches to these problems [50], and as such, it is hard to truly compare methods or to see how much progress is being made in the field. Currently, a variety of accuracy metrics are used across the literature in the area. However, a comparison of different algorithms and metrics against each other and also against an industry model finds that the metrics cannot accurately capture what users like [41]. Most current approaches to evaluation [29, 40, 45, 61] are treated as document retrieval problems, and measure how well algorithms can predict songs that are used in ground truth, or pre-existing, playlists. Some other approaches use limited user studies [30, 43], listening log analysis [30, 26], or their own combination of the above [52]. They do not, however, tend to explicitly take playlist quality, user preference, or playlist purpose into account. Some proposed methods for evaluation that come closest to this [50, 36, 52] focus only on a singular measure of user enjoyment or playlist quality, such as diversity [36], or similarity [52], and are not necessarily applicable to general settings outside of their research [52]. This project is based on the belief that multiple measures and a more general approach to evaluating

playlist quality can be proposed, to help bridge the gap between mathematical and user evaluation.

To propose an idea for this metric, I will examine the current state of the APG and APC field, specifically how seminal, influential, and modern approaches to both problems are evaluated, and also collect information from previous preference elicitation research on playlist creation and consumption, to identify measures of playlist quality and user enjoyment. I will then choose, with justification, some key measures to formulate and combine to form the basis of the new metric. The metric will then be evaluated on a small user study, in order to understand whether the measures could reasonably capture user enjoyment and playlist quality.

In Chapter Two, a brief outline of the field of APG and APC are given. Chapter Three then focuses more specifically on how APG and APC are being evaluated, details a variety of metrics used frequently across the literature, and analyses the issues with these techniques. Chapter Four details the proposed new playlist quality metric, explaining what measures have been chosen and why, as well as their formulations. The metric is evaluated in Chapter Five, via a small user study, which is detailed and results reported. Finally, Chapters Six discusses limitations of the project, suggestions for future work in the area, and conclusions are drawn about the project and the proposed metric.

1.1 Contributions

The main contributions of this project are as follows:

1. A comprehensive overview of evaluation techniques and metrics used across academic approaches to Automatic Playlist Generation and Continuation, including how, and where they are used, and discussion of why they are not ideal for evaluation of these problems.
2. A proposed new evaluation metric for recommended playlists that takes into account different ‘objective measures’ of playlist quality, to more accurately meet user preferences.
3. A user study undertaken on the new proposed metric, which shows a good starting point for future work on better playlist evaluation, as well as solidifying some ideas in previous research about user playlist curation habits, and introducing some more attitudes on the topic.

Chapter 2

Background

This chapter explores in more detail the problem of recommending playlists, specifically the defined problems Automatic Playlist Generation and its sub-problem Automatic Playlist Continuation. It outlines the motivation behind using playlists as a recommendation item, defines the specific recommendation problem being looked at, and explores some of the main approaches to this problem, as well as touching on how they are evaluated.

2.1 Digital Music Consumption & Playlist Usage

In the digital age, streaming services are one of the main ways people consume music, with Spotify for example amassing 299 million users [19]. One of the main selling points of these services is the personalised approach. Spotify creates two weekly playlists and six 'daily mixes' for each of its users, as well as regular one-off playlists [16]. The platform hosts over four billion playlists [19], and studies have identified that playlist consumption and creation are an important way users interact with music in the modern era. The Music Business Association identified in 2016 as part of an insights program that in the USA, 31% of users listening time was spent on playlists, more so than albums [14]. MIDiA conducted a study that found that 55% of streaming service subscribers created playlists [18]. Nielsen in 2017 identified that in the USA, 58% of users created their own playlists and 32% shared them with others [13]. It is clear from these studies that users of music streaming services are growing and their consumption of playlists, as a result, is too. Since the conception of digital music platforms, thought has been given to the idea of Automatic Playlist Generation (APG), with researchers such as Platt [54], Patchet [51, 23], Tewfik [20, 21], and Logan [46] proposing early solutions to the problem. Schedl et al. [57] identify APG and the related issue of Automatic Playlist Continuation (APC) as one of the major developments and challenges in music recommendation in recent years, due to the benefits it has for these streaming companies. Having the ability to extend or create playlists keeps users using streaming services for longer sessions, and adds to the personalisation of the service, which if increasing customer happiness, will potentially keep the user with their platform for longer [57].

2.2 Playlists

Songs as recommendation items differ to other typical recommendation items such as books or movies, due to the different way they are consumed by users. Songs are typically far shorter than movies and books, for example. They are also mainly consumed in a session, back-to-back, rather than one at a time [62]. It is useful, then, to use playlists as a recommendation item, instead of single songs.

A playlist, simply, is a collection of songs created by a user, varying in length, and fulfilling some purpose or theme, depending on factors such as the user, the target listener, or the environment to be consumed in. For example, Cunningham and Bainbridge [33], in their user study on playlists and mixes, find that playlists tend to fall into different categories such as musical style, event/activity, or mood. More simply, a playlist is a collection of songs designed to be listened to together. Early in the APG field, Tewfik et al. [21] specifically define a playlist as:

A sequence $S = v_1, v_2, \ldots, v_n$ where variables v_i are songs taken from a collection or database.

2.3 Automatic Playlist Generation

Automatic Playlist Generation was first proposed and implemented by Patchet et al., and Tewfik et al. [51, 20] in the early 2000s as a means of 'Electronic Music Distribution'. As digital music consumption was increasing in popularity, researchers were trying to propose ways to best recommend music to users. Tewfik et al. [20] identified that playlists are a useful way to recommend music to users as there can be issues with buying whole albums - when the listener may only want one or two songs. The task of Automatic Playlist Generation (APG) is generally the task of taking one or more seed songs and creating an entire playlist based on them. Bonnin et al. define in a 2014 survey on playlist generation techniques [25] the challenge of automatic playlist generation as follows:

Given (i) a pool of tracks, (ii) a background knowledge database and (iii) some target characteristics of the playlist, create a sequence of tracks fulfilling the target characteristics in the best possible way.

Which is a broad definition covering both automatic playlist generation as well as the more specific sub-task of Automatic Playlist *Continuation*, covered in the next section. The definition of APG defined above is the one we will be referring to for the rest of the project.

Tewfik et al.'s seminal works on APG [20, 21] are based on a network flow model. They take a start and end song, and using similarity measures, find a path of user-defined length from the start to end songs. It minimises the difference in the songs based on attributes given for each, such as genre, tempo, or length. In [21] they improve their initial approach to add additional generation cases - having only the start song, only the end song, or neither. However, the model was criticised at the time by Aucouterier et al. [23], for not scaling up to deal with massive databases, which

is even more relevant in the current context of digital music. Acouterier et al. [23] instead focuses on an adaptive search method based on song metadata, such as tempo and genre. Their approach aims to find a playlist as a path satisfying some constraints, the example given being a playlist of songs with increasing tempo, with half of the songs belonging to the Folk genre, and half belonging to Rock.

Other influential works in APG approaches include that of Logan [46], who took a similar approach to Tewfik et al., mapping a song library to a graph, and attempting to find the shortest path from a seed song to create a playlist. The distance measure between songs, however, is based on audio content instead of metadata. Logan's approach is evaluated on a count of relevant songs retrieved at various points across the playlist, which is also known as R-Precision (Section 3.5.2).

Some more recent approaches to APG include that of Ben Fields [35], who proposes a novel similarity measure for songs and then uses path-finding to create playlists based on this similarity measure; Liebman et al. [45] who use reinforcement learning techniques based on user song and song transition preferences - an approach also used by Chi et al. [30] in 2010 where they base their model on emotions; Irene et al. [40] who base their playlist recommendation model on a mix of Recurrent Neural Networks (to model sequences based on pre-existing playlists) and Convolutional Neural Networks (to learn audio descriptors of songs); Pichl et al. [53] who combine metadata such as playlist names and user-specific information to cluster playlists, then perform collaborative filtering on the clusters to classify and generate new playlists; and Chen et al. [28] who construct user-song-playlist graphs (that is, the relations between a user, their music library, and their playlists) and then use path-finding algorithms to recommend a playlist to a user.

These approaches, excepting Liebman et al. [45] and Chi et al. [30], are all evaluated using an Information Retrieval (IR) style technique, where the systems are trained and tested on existing datasets of playlists, and their success is based on their ability to recommend missing songs correctly, i.e., retrieve the 'relevant' documents. Liebman et al. and Chi et al. instead utilise a more user-centric evaluation technique, with Liebman et al. simulating 1000 users based on clustering existing playlist data, and predict their enjoyment of generated playlists. Chi et al. do this on a smaller scale, simulating two different types of user, as well as evaluating on a user study.

2.4 Automatic Playlist Continuation

Automatic Playlist Continuation has gained traction in recent years as a sub challenge of playlist generation [57] and was the focus of the ACM RecSys challenge in 2018 [29], supported by Spotify. Many of the papers proposing and implementing methods for APC come from entries to this competition, so they will be discussed here. APC is referred to as a sub problem of APG [57], where instead of generating a full playlist, the system should recommend tracks to add to the end of an already created or partially-created playlist.

2.4.1 ACM RecSys Challenge 2018

In the RecSys challenge 2018, the contestants were provided with a Spotify dataset with a million playlists that they could train their models on. They were to then predict for a playlist the 500 top tracks (ranked in order of most likely to least likely) that should be added next to the playlist [29].

The team placing first in the competition proposed and implemented a two stage model [64] for APC recommendation. In the first stage, a combination of collaborative filtering, Convolutional Neural Networks, item-item and user-user models are used to recall a large volume of next-song recommendations. The results from this are then fed into the second stage, which re-ranks the songs. They performed the best out of the entries to the competition based on the Information Retrieval (IR) based withheld tracks procedure, where the algorithms are evaluated on their ability to recommend missing tracks from the playlists in the dataset. The metrics chosen by the organisers for this evaluation were R-Precision, Normalised Discounted Cumulative Gain (NDCG), and 'Clicks'.

Other approaches to APC in the competition used similar multi-stage models, combining different machine learning techniques to build their systems. Matrix Factorisation Collaborative Filtering was also employed extensively by contestants, as well as neighbourhood-based systems, all achieving good results according to the organisers. [66]

2.4.2 Other Approaches

Other approaches to APC outwith the scope of the RecSys Competition include that of Vall et al. [62], Gatzioura et al. [36] and Tran et al. [61].

Both Tran et al. [61] and Vall et al. [62] propose systems based on the typical collaborative filtering approach to playlist recommendation, modeling the relationship between users-playlists, and playlists-songs. Tran et al. [61] then add a feature for song-song similarity, but their main idea in this approach is that their similarity measure is based on the Mahalanobis distance, rather than a dot-product vector similarity. Vall et al. adapt their approach for 'out-of-set songs', that is, songs that don't exist in the playlists used in their training sets, by learning the probability of an out-of-set song being added to each playlist. Both approaches evaluate using the IR technique, with metrics Hit Rate / Recall, and Tran et al. also use NDCG.

Gatzioura et al. [36] suggest a hybrid recommender system for APC, focusing on recommending 'similar concepts' for playlist continuation, rather than necessarily similar specific items or users. They combine techniques more commonly applied to natural language processing problems such as Latent Dirichlet Allocation (a topic modelling technique), as well as case-based reasoning. In this approach, the authors treat a playlist as a session, and focus more on human reasoning than typical recommendation approaches, due to the 'semantic gap' idea in music recommendation [27].

2.4.2.1 The Semantic Gap

The Semantic Gap, referred to occasionally throughout the rest of this project, is the idea that there is an inherent difference between the way computers and humans describe things [58]. This is particularly relevant in the area of music recommendation, and is highlighted as a key issue by Gatzioura et al. [36] in many approaches to music recommendation. They discuss how humans and recommender systems tend to represent and describe music very differently, with humans more focused on moods, feelings evoked, and styles, whereas machines can only go so far using concrete pieces of information like artist, tempo, or song length, for example.

2.5 Session-Based Recommendation

Another area of research that can be applied to APC is that of session-based recommendation. These types of recommendations are increasingly used in commercial settings to predict a user's next action, or click on an item, depending on their previous actions in the current 'session', or use of the commercial service [39]. First proposed in 2015 by Hidasi et al. [39], they thought to apply deep learning techniques to recommendation settings after Neural Networks had proven successful in other areas.

Vall et al., for example, in [63] evaluated four different models on playlist recommendation, including an RNN. They found that the RNN clearly outperformed the other, simpler models they used as baselines. RNNs also partially form the basis of the approach of Irene et al. [40], who combine the sequence based modelling of RNNs with Convolutional Neural Networks to also learn audio descriptors of songs.

2.6 Evaluation Techniques

What all of these approaches, of course, have in common is that they all need to be evaluated in some way. Due to issues such as the Semantic Gap, detailed above, capturing user enjoyment in recommender systems is a key problem in the field [25].

Human evaluation, via techniques such as user study, is the best way to evaluate any type of recommendation on real target users of the system. However, this is extremely time consuming and costly [25], especially in an area such as music where so many songs, consumers, and various digital platforms exist. Instead, most of the approaches seen in the literature so far rely on accuracy measures like those used in other machine learning and information retrieval problems. There are also some other creative techniques such as user listening log analysis that have been used sparsely across the field [26, 30]. The next chapter contains an analysis of the current field of APG and APC evaluation techniques and metrics, where these ideas are discussed in more detail.

Chapter 3

Evaluation of Playlist Recommendation

3.1 Recommender Systems Evaluation

Recommender Systems are typically hard to properly evaluate, because of the human element of recommendations. The best, and most preferred way of evaluating a recommender system would be by human evaluation [25], but sourcing a wide enough range of people to make any evaluation concrete is difficult. This becomes even harder on a larger scale project, such as some modern music recommendation systems which use datasets of hundreds of thousands of playlists or songs to train their approaches. This leads to academics using a variety of evaluation metrics or procedures to try and gauge the success of their approaches to these problems. However, this causes a lack of consistency across different academic approaches to these recommendation problems, and cannot always accurately capture human preference [50].

3.2 APG & APC Evaluation

McFee and Lanckriet [50] identify that specifically in the field of playlist recommendation, there is not currently a standard evaluation procedure for testing playlist generation approaches. This makes it hard to accurately compare approaches or to see how the area is progressing over the years. Since there isn't a specific way to evaluate playlist recommendations, two seemingly well performing approaches that have been trained or evaluated using different techniques, when both compared on one specific evaluation technique, may actually perform very differently, and perhaps be outperformed by another approach that does well with this other chosen measure. When newly proposed approaches are also perhaps evaluated differently to older approaches, there isn't a clear way to see if real improvements are being made.

Bonnin et al. [25], in their survey of playlist generation, identify that this evaluation of automatically generated playlists falls into four main categories: user study - human evaluation, where participants are asked for their opinions on the generated playlists;

listening log analysis - where human participants are recommended playlists and their listening habits are monitored after the fact to see if the recommendations changed what they were listening to; objective measures, which are measures of playlist quality, for example, how enjoyable the songs in a playlist are when all listened to together (cohesion); and comparison to hand curated playlists, which uses pre-existing datasets and Information Retrieval (IR) evaluation metrics to analyse how well algorithms can recommend ground-truth songs. Each of these evaluation techniques have benefits and drawbacks. The four main categories are described below in more detail, as well as citing some approaches in the literature that use these techniques. A fifth approach of user simulation, which has been emerging in recent years, has also been added, as well as specific details of two proposed techniques for evaluating playlists - McFee and Lanckriet's natural language based model [50] and Pauws et al.'s PATS [52].

3.2.1 User Studies

User studies, as discussed above, are the ideal way to evaluate any type of recommendation system, as they allow us to assess the real perceived quality of recommendations. However they are time consuming and costly, and as Bonnin et al. [25] identify the study size is often very limited, especially in the field of APG as it currently stands. Knees et al. [43], for example, evaluate their proposed method for generation over a user study of only 10 participants.

These user studies typically involve presenting participants with a generated playlist, either specifically created for them [52], chosen by them from a set of available playlists [44], or randomly generated [50]. The participants are then interviewed or asked about the playlists, and may be asked to rate them [43, 52]. Knees et al., for example, provide the same 10 playlists to each of their 10 participants, and the average rating across the participants for each playlist is found. A more specific type of user study involves analysing the participants' listening activity after recommendation, which is discussed more in Section 3.2.2.

User studies in the literature are currently mainly used for preference elicitation. Studies such as that by kamalzadeh [42], Lee [44], Andric and Haus [22], and Cunningham [33] all explore playlist creation from a user perspective, and discuss what their participants look for in playlists.

3.2.2 Listening Log Analysis

Listening Log Analysis is a specific form of user testing. Instead of directly asking users for their opinion, they are given recommendations generated by an automated system, and their listening behaviour is then monitored for a period of time after the recommendation has taken place. The perceived success of the recommendation is then measured by how much it has affected the user's listening behaviour.

Bosteels, Pampalk, and Kerre used this approach in their paper [26], stating as their motivation a disagreement with the use of Information Retrieval and Machine Learning techniques for evaluation of 'dynamic playlist generation', their defined problem at the time. It is also more recently used by Chi et al. in their reinforcement learning

approach, where they monitor user behaviour and feedback such as skips, replays, and likes [30].

3.2.3 Objective Measures

Objective measures, as identified by Bonnin et al. [25], describe specific criterion for playlist quality, such as homogeneity - how cohesive a playlist is and how similar its contained songs are; variety/diversity - how much of a mix the playlist contains, whether of artist, genre, time period, or some other chosen characteristic; freshness or how 'new' the tracks are; or smoothness of track transitions.

Papers such as [36] utilise the objective measure approach to aid in evaluating playlist quality. However, usually only one criterion is focused on in specialised research, such as by Slaney and White [59], who focus entirely on diversity, specifically in this approach the mix of genres in playlists, or Lee's user study [44] which aims to find out about similarity, or Ward et al.'s research into the effect of song familiarity [65]. Bonnin et al. [25], and Fields [35] identify that combining multiple of these measures is something that needs further explored in the area, and working out how to balance for different criteria - *"Therefore, more comprehensive evaluation approaches are required that use multiple measures at the same time"* [25]. This idea is what the proposed metric of this project is based on.

3.2.4 Comparison with hand-curated playlists

In the literature currently, the most popular evaluation technique is comparison with hand-curated playlists. This involves a system being trained to generate or continue playlists that already exist. They can then be evaluated on how well they recommend or predict the songs that the real playlists contain. This technique uses a variety of evaluation metrics commonly used in Information Retrieval (IR) problems, such as Normalised Discounted Cumulative Gain (NDCG), or Precision, to evaluate how well the recommender systems retrieve the missing songs.

This approach is used as the standard evaluation technique in the 2018 ACM RecSys competition on Automatic Playlist Continuation [29] with the metrics NDCG, R-Precision, and a defined measure known as 'Clicks', as well as in other approaches to APC, for example Precision and F-Score in [36], and NDCG and Recall in [61]. It is also used widely across approaches to Automatic Playlist Generation: Precision and Recall in [41]; Error Metrics (Mean Absolute Error, Root Mean Squared Error, and R^2 Error) in [40]; Precision in [28]; R-Precision in [46]; Precision and a 'Coverage' measure defined by the authors in [52]; and Precision and F-Score in [53].

These metrics are identified by some authors in the literature as too rigid to capture true user enjoyment [36, 26]. They can only capture playlist quality in terms of ground-truth playlists, but do not consider that there may be other recommendations that different users may prefer. The metrics are discussed in more detail in Section 3.5, as well as their issues in Section 3.6.

3.2.5 User Simulation

Outside of the four main areas identified by Bonnin et al., [25] there is also an emerging technique of using user simulation for playlist recommendation evaluation.

The idea is utilised by Chi et al. [30] as well as Liebman et al. [45], where they create simulations of users and attempt to recreate how they would react to their proposed systems. In [30] they create two simulated users to fine-tune their algorithm on, each with different described behaviours and preferences. In [45] they generate 1000 simulated listeners based on clustering of playlists from a dataset, then take samples from them and feed their expected behaviour into the training model.

This approach is an interesting way to recreate human preference on a wider scale than would be possible with real users, however may be computationally costly, and more research would certainly need to be done into how accurately the simulated users can represent real human opinion.

3.2.6 McFee and Lanckriet's Natural Language of Playlists

McFee and Lanckriet's proposal for a novel playlist evaluation technique [50] focuses more on ideas in natural language processing, instead of information retrieval. They model playlists as collections of words (songs) belonging to an unknown language, and then use different similarity measures and learning procedures such as clustering and Markov chains, to learn the probability of a generated playlist occurring naturally, based on existing datasets.

They conclude that their technique is effective at evaluating playlists, and they run it over two different academic approaches. However, what isn't explored in the paper is whether the proposed technique accurately captures user enjoyment, just that it can predict naturally occurring playlists.

3.2.7 Pauws and Eggen's PATS: Realization and User Evaluation of an Automatic Playlist Generator

In Pauws and Eggen's proposal for APG [52], they perform an intricate evaluation technique with a combination of IR metrics and user study. Their system, PATS, is based on a user-specific value of coherence vs diversity, where the user is asked questions about what they like in a playlist for different contexts, these ideas are mapped to a similarity measure, and songs are clustered together, then recommended appropriately to the user. The system outperforms their baseline, and indicates that this type of quality measure reflects user enjoyment.

However, the study focuses entirely on the genre of Jazz, and so it is not necessarily the case that their findings can be generalised across more music. In addition, despite the perceived success of their evaluation technique, they used novel software and a specific study environment, which means the evaluation is not reproducible elsewhere in the field [52, 25].

3.3 'Good' Playlists

In recent years, research has been done via user study and data analysis into what users want from playlists, in other words, what makes a playlist good or bad. Understanding this is key in evaluating automatically recommended playlists and managing to capture what increases user enjoyment.

Some of the studies referenced extensively throughout the rest of this project include Kamalazedah [42], Lee [44], Cunningham and Bainbridge [33], Hagen [37], Andric and Haus [22] , Sarroff and Casey [56], Hansen and Golbeck [38] and Ward [65].

The main findings from the studies include that familiarity of songs to users has a strong impact on their perception of playlists [42, 44, 65]; that many users like a playlist that is cohesive and similar in sound while also valuing a good mix of artists and diversity so that they don't get bored [42, 22, 44]; that smoother transitions between songs makes for a more enjoyable listening experience [56, 33]; that choice of start and end song can be important to users [38, 33]; and that the chosen purpose of a playlist is the most important indicator of its content [33].

In Cunningham and Bainbridge's influential study on playlist curation behaviours [33] conducted on six in-person interviews, seven online interviews, 24 posts, and 115 threads on the Art of the Mix playlist discussion forum, they find that a playlist's purpose is a key indicator of what content it should include. These purposes, defined by them, roughly fall into six main categories: artist, genre, musical style, event/activity, semantics (e.g. 'Romance'), and mood. For the participants in the study, these differing purposes greatly changed what they valued in the playlist. If a user creates a playlist based around a genre, Jazz, for instance, and then is recommended a Rock song, they are likely to be dissatisfied with the recommendation. However, if both songs were love songs and were recommended for a playlist based around romance, they could be seen as good recommendations.

The purpose of a playlist is thus key to understanding which recommendations are good and which are bad. This comes in to play in an evaluation style based on pre-existing playlists. As Lee [44] points out, since users perceive music in different ways a playlist being good is very subjective to a specific user. While some may deem songs similar (and thus want to include them in the same playlist) because of lyrical content, for example, others perhaps look for similar instrumentation. So while a generated playlist may match some pre-existing playlists containing the seed song or songs, the user being recommended for might have had an entirely different purpose in mind.

Because of this highly subjective perception of music and playlist purpose, in a more comprehensive measure of playlist quality, features such as familiarity, cohesion, variety, or track transitions must be taken into account. Further than that, the desired level or balance of these measures should be personalised for each user to account for their preferences and purpose in creating playlists.

The next sections discuss the use of information retrieval metrics commonly found across the literature on APG and APC, as well as issues these metrics face when used for evaluation.

3.4 Withheld Tracks Procedure

In most of the applications of the metrics discussed below, they are used within the technique of withheld tracks. For context, this procedure involves the use of some playlist or song dataset, where songs or playlists are deliberately withheld by the researchers. The algorithms' effectiveness are then based on their ability to recommend the ground truth missing tracks. This evaluation can be done using a variety of different Information Retrieval (IR) metrics. These metrics, as used by approaches to APG and APC across the literature, are explored below.

3.5 Evaluation Metrics

3.5.1 Normalised Discounted Cumulative Gain

Normalised Discounted Cumulative Gain (NDCG) is a successor to Discounted Cumulative Gain (DCG) and Cumulative Gain (CG). It is a measure of 'ranking quality' for a recommendation set from an IR system, based on the assumption that the more relevant a document is, the more useful it is to the user [12]. This metric is used in approaches to APC [29, 61, 66], where the recommendation set is the song suggestions made to the user to add to their playlist.

Each item in a recommendation set is given some relevance score, and CG is simply a sum of these scores. The higher the CG, the more relevant the recommendation set is in terms of this relevance. It is calculated as follows [4]:

$$CG = \Sigma_{i=1}^{n} relevance_i$$

DCG, as a succession of CG, also accounts for ranking position, giving a higher weight to more relevant documents being ranked first, and can be calculated one of two ways:

$$DCG_1 = \Sigma_{i=1}^{n} \frac{relevance_i}{log_2(i+1)} \qquad \text{or} \qquad DCG_2 = \Sigma_{i=1}^{n} \frac{2^{relevance_i}-1}{log_2(i+1)}$$

Where DCG_2 more heavily penalises cases where highly relevant documents are ranked lower. NDCG then takes either of the calculations for DCG and adds a final consideration, that recommendation sets can vary in size across users and recommendations. NDCG normalises DCG for these sets to a range [0,1] so that all recommendation sets can be compared. It takes the DCG of a set over the ideal DCG (iDCG), that is, one where the ranking decreases as relevance decreases.

$$NDCG = \frac{DCG}{iDCG}$$

3.5.2 R-Precision

R-precision is another metric also used in IR problems, and is applied to APG & APC in [29, 46]. R-precision is a measure of how many relevant documents are retrieved by a system at a specific number of retrievals R. That is:

$$R - Precision = \frac{r}{R}$$

where r = documents that are relevant **and** retrieved and R is all documents that are known to be relevant until a certain level [31]. In the missing tracks evaluation technique, the 'relevant documents' are the withheld songs.

3.5.3 Precision

Precision is utilised by some playlist recommendation approaches like [36, 41, 53, 28], usually in combination with another metric such as Hit Rate/Recall, or within the F-Score, both discussed in the next sections. It calculates the proportion of retrieved documents that are relevant, which in the case of APG and APC, are the tracks that are withheld. It follows the formula:

$$\frac{RelevantDocs \cap RetrievedDocs}{RetrievedDocs}$$

The key issue with precision and recall (Section 3.5.4) is that they are dependent, and increasing one tends to decrease the other [15]. Since precision only accounts for the rate of relevancy over all documents retrieved documents, it does not care for how many of those documents were actually retrieved, i.e. a system that retrieves only three documents, that were all relevant, would perform extremely well on precision, but those three documents may be from a pool of say 100 documents that *should* be retrieved. In APC or APG systems that use the missing tracks evaluation procedure, the system would be performing well on precision by correctly suggesting a small number of the hidden tracks, but may be missing many others.

3.5.4 Hit Rate / Recall

Recall, alongside precision, is a classic accuracy measure used in information retrieval and machine learning problems. In the APG and APC literature, it is also often referred to as hit-rate, as it is calculating a proportion of tracks predicted correctly, i.e. how many the algorithm 'hits'. Hit-Rate is used in approaches such as [41, 36, 61, 62]. Recall, conversely to precision, calculates how many of all retrieved documents are relevant. It is formulated as [15]:

$$\frac{RelevantDocs \cap RetrievedDocs}{RelevantDocs}$$

In practice, recall has issues in the same way that precision does, with the fact that the two measures depend on each other, and an increased recall score means reduced precision. Specifically, when using recall, the volume of retrieved documents is not taken into account. This means that a system that just recommended all existing songs would perform very well on recall, but this is clearly not practical.

3.5.5 F-Score

The F-Score or F-Measure, is a common way of combining and balancing precision and recall in evaluation metrics. The F-Score is an unweighted harmonic mean of both precision and recall, calculated as [6]:

$$\frac{2.Precision.Recall}{Precision + Recall}$$

F-Score is used in some APG and APC applications instead of relying on either precision or recall, such as [36, 53]. In Gatzioura et al.'s approach, they adapt the F-Score to be the harmonic mean of precision and playlist diversity, instead of precision and recall.

The F-Score occasionally faces criticism, for example for the fact that it gives precision and recall equal rates, whereas in practice, one is likely to be more important than the other, or that it doesn't take true negatives (TN) into account, which may be important for some applications [55].

3.5.6 Spotify Recommended Song Refresh Clicks

Clicks is a measure specifically applied to APC within the 2018 RecSys competition [29] and is based on a Spotify-specific feature, so a clear issue with this as a metric is the fact that it doesn't widely or generally apply to evaluation of this problem.

Spotify's 'recommended songs' feature appears at the bottom of user-created playlists [16], and displays 5 or 10 songs that the user may wish to add to the playlist. The songs can be added or ignored, and the list can be refreshed by the user to see new recommendations. Clicks, in the context of the RecSys competition, is the average number of times a user would need to refresh the recommended songs using the contestants approach until the first time a withheld song is recommended.

A similar idea to clicks is used in Chi et al.'s reinforcement learning approach to APG [30], who perform listening log analysis and monitor various user inputs. One of these is the 'Miss-to-Hit Ratio', which calculates that from a recommended playlist, how many times the user has to skip recommended tracks to find one they like or want to listen to.

3.5.7 Other Measures

There are some other evaluation metrics utilised by papers mentioned in this report that are not necessarily common across the rest of the literature. A brief description of some of these is given here.

3.5.7.1 Error Scores

Error measures are utilised in Irene et al.'s [40] approach to APG to measure how closely their hybrid approach can predict real-life playlists. The playlists are represented as vectors with their features being songs, and the error is measured as the difference between the generated output and the ground truth playlist. They use the R^2 Error, Root Mean Squared Error $RMSE$, and Mean Absolute Error MAE .

3.5.7.2 Coverage & Ratings Score

Coverage is a measure defined specifically by Pauws and Eggen in their PATS evaluation [52]. It is calculated as the cumulative number of preferred songs in each successive recommended playlist that were not present in previous playlists. In their defined

ideal scenario, this would approach the number of songs in all user playlists, indicating full coverage of a users preferred material via playlist recommendation.

Pauws and Eggen also use a human-defined metric which they term 'ratings score' [52]. The score is simply what a user rated a playlist generated for them, on a scale from 0 = 'extremely bad' to 10 = 'excellent'.

3.6 Criticism of Current Evaluation

With the existing evaluation techniques, specifically with the use of IR metrics as detailed above, there are key issues in how well they can capture user preference.

In some of the more recent approaches to playlist recommendation such as [36, 26, 30], there is a distinct move away from the use of IR metrics, with both [36, 26]'s authors citing their reasoning that they are too strict to capture user enjoyment. This idea is expanded on by both Gatzioura et al. [36] and Bonnin et al. [25], who identify that the IR technique can only evaluate how well a system recommends specific relevant songs based on the pre-existing playlists, whereas these may not be the only good recommendations, or even the best. In Gatzioura et al.'s paper [36] they choose to combine their use of IR metrics such as precision, with quality-based criterion of coherence and diversity, as well as a distance measure between recommended and ground-truth playlists. Their choice to explore so many different evaluation techniques could indicate demand for a new, more general approach to the evaluation of generated playlists.

Alongside this, these metrics only measure how closely the systems can recommend based on the ground truth, but the playlists being generated by the algorithms aren't tested for enjoyment in their own right. Quality, in any form, is not explicitly being taken into consideration, but instead the evaluation relies on the assumption that the metrics implicitly take into account objective measures from historical playlist data.

In fact, in a comparison between academic approaches to APG and an industry standard, Jannach et al. [41] find a clear gap. They evaluated different academic approaches like K-Nearest Neighbours, Content-Based machine learning, and 'Collocated Artists Greatest Hits' (which recommends popular songs by similar artists as the seed artists) against The Echo Nest (TEN) [3] on the same metrics to compare across approaches. The Echo Nest was acquired by Spotify in 2014, and its approach to song recommendation has been fine-tuned through years of A/B user testing. The study found that academic approaches followed a similar mix in their generated playlists of measures such as popularity and diversity, all with a heavy skew towards popular artists and songs compared to the industry standard, as seen in *Figure 3.1*. They also found that the academic approaches tended towards increased track diversity compared to TEN. Since TEN's approach was evaluated and improved based on user testing, there is a strong indication that academic approaches do not reflect real user preference.

This heavy skew towards popular tracks and artists in academic approaches could be down to the choice of evaluation. Specifically, when using IR metrics such as those detailed in Section 3.5, the performance of the algorithm is based on a measure of relevance. In IR problems, where the system may be fetching documents, relevance

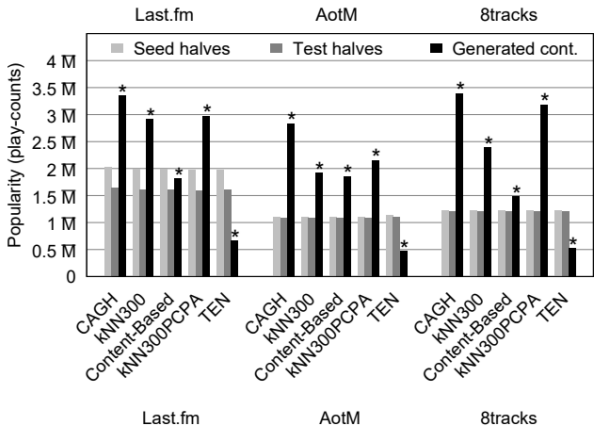

Figure 3.1: popularity across approaches to playlist generation [41]

is easier to capture through analysis of key-words and themes. This is far harder to model for an area as diverse and subjective as music. When the measure of 'relevant documents' is based on pre-existing playlists, the algorithm will reflect choices that apply to a generalised view of users. If the algorithm is trained to recommend songs that appear in historical playlists, there will be a natural skew towards the songs and artists that appear more often.

Additionally, as McFee and Lanckriet [50] identify, there is a lack of consistency in evaluation approaches across the APG and APC field. Since there isn't a dedicated metric or technique for specifically evaluating playlists, authors are free to choose from a wide variety of metrics when analysing their results. This hinders progress in the area, as approaches cannot always be directly compared without some further analysis, and it makes potential improvements through the years hard to track. There is also potential for a biased choice of metric by academics, as a metric or technique that reflects best upon their proposed approach may be chosen because of a lack of standard practice.

Other approaches to playlist evaluation outside of IR metrics are commonly limited by time and resource. User studies, including Listening Log Analysis, cannot be relied upon to evaluate large-scale Recommender Systems, even though they are most accurate at capturing human interest. Basing the success of an entire system on a small number of users does not allow results to be generalised across hundreds of thousands of songs, users, and playlists. Current evaluation techniques that do take into account objective measures do not account for the various different ways playlist quality can be measured, instead usually focusing on one quality measure such as similarity [52, 36]. Other, more novel approaches, like the probability-based evaluation used by McFee and Lanckriet [50], similarly to IR metrics, are entirely based on pre-existing playlists, and do not allow for the possibility that there may be other, better recommendations than what has already been created. I believe it is possible to create an evaluation metric that instead accounts for isolated playlist quality, based on a combination of objective measures explored previously in the field, and user preference.

Chapter 4

Proposed Metric

In this chapter, I propose a new metric to be used for evaluation of automatically generated playlists. The metric is a combination of three main objective measures, chosen for their importance based on previous research into playlist creation and consumption. I outline the motivation for choosing each of these measures, the representations for each, and their combination and ideal use in practice.

4.1 Aims

The proposed new metric is designed to be a measure of isolated playlist quality, and more accurately capture user-specific preferences than the strict Information Retrieval (IR) metrics currently most popularly used for algorithmic evaluation, which have issues as described in Section 3.6. It does this through calculating key objective measures that indicate specific playlist quality criterion.

4.2 Chosen Objective Measures

In their 2014 survey of Automatic Playlist Generation, Bonnin et al. [25] identify 'objective measures' as one of the key ways algorithmically generated playlists can be evaluated. This idea has been explored in a variety of ways across the literature, for example by Fields [35] who explores song similarity, by Slaney and White [59], and kamalzadeh [42] who explore playlist diversity, and by Ward et al. [65] who explore the effect user familiarity has on listening behaviour and recommendation. However, rarely are these measures used in combination with one another for evaluation purposes. Gatzioura et al. [36] use song similarity, diversity, and IR metrics such as precision to evaluate their proposed approach for APC, however do not take into account, for example, user familiarity. Pauws and Eggen [52] use a similarity measure in their proposed evaluation technique PATS, and discuss it as being the opposite of diversity, thus combining the two but not taking variety or other objective measures explicitly into account. Further research into the combination of the criterion is mentioned by Bonnin et al. [25] as a necessity in the field, as well as an idea described by Fields [35].

My proposed metric for playlist evaluation takes into account three main objective measures: user familiarity, playlist cohesion, and artist / genre variety. Hereafter, the three are referred to simply as familiarity, cohesion, and variety. These objective measures have been chosen to combine and create a metric for playlist quality due to their frequent mention in existing literature on playlist consumption as something that users take into account in their own playlists [44, 33, 42, 37], or as something that affects users listening choices [65]. Their definitions and formulations are discussed in the next sections, alongside a more detailed look at the motivation for why they have been chosen to represent playlist quality.

4.2.1 Familiarity

Familiarity is simply a measure of how much of a generated playlist a user already knows. This familiarity of songs to users has been shown to affect their opinion of a playlist in studies such as by Lee [44], kamalzadeh [42], and Ward [65]. For example, in [44], having one song that a user had a pre-defined opinion of had a strong correlation to their enjoyment of the playlist as a whole. Lee even goes on to state that participants' *"evaluation of playlists tended to be quite subjective as they were highly affected by personal preference and familiarity with the music on the list"*.

In a larger user study ran by kamalzadeh [42] on 222 participants, which was a study into listening habits as well as curation behaviour, almost 100% said that they preferred familiar songs to listen to over new songs. This is a strong argument for the importance of including familiar songs in generated playlists. Of course, since the purpose of a playlist may be discovery, it doesn't necessarily follow that these same participants would never listen to new songs or include them in their playlists. What is important is finding the balance for each user. Ward et al. [65] also explicitly explore the power of familiarity on user listening behaviour, based on user music tastes and choice of radio station. They find that predicting a user's familiarity with certain music styles greatly increases the chance of predicting what radio station they will choose to listen to, and conclude that familiarity can be more of a solid indicator for user listening habits and enjoyment than even their own stated liking of music.

Having songs, or at least some artists, that the user is already pre-disposed to like can increase their satisfaction with a playlist, as well as give them a framework to explore new music within. Such is the opinion of a participant in Lee's study [44], for example, who says *"A mix of things is good, cause I would like to discover new artists, it's always a good way to (be) introduced through [...] artists you already like"*. The base idea, of course, that taking user preferences into account when making recommendations is not novel, being the basis for Collaborative Filtering based Recommender Systems [1]. In the proposed metric, this idea is expanded instead to a measure of playlist quality and combined with the other objective measures chosen.

When defining the formula for familiarity across a playlist and a user, inspiration can easily be taken from some of the IR metrics discussed in previous sections, which rely on 'relevant' documents to be calculated. One of these measures is NDCG, described in more detail in Section 3.5.1. NDCG can be adapted for use in the familiarity measure where the relevant documents are the songs and artists that are known (and liked) by

the user being recommended for. A similar approach is used in the ACM RecSys Competition 2018 [29], where submitted algorithms are evaluated on their ability to predict ground-truth songs or artists. As discussed in Section 3.6, using this for evaluation has its own issues, but the metrics do work for their typical purpose of document retrieval, so the general idea can be altered appropriately to calculate familiarity instead.

Firstly, since familiarity is more binary than relevance, i.e. a song is either known to a user or not, the familiarity measure does not need to take order into account in the way that NDCG does. We want to calculate the overall familiarity of a playlist as a whole, and thus having familiar tracks near the start of the playlist does not make a difference. Andric and Haus [22] found that when songs are familiar to users, their overall order doesn't tend to matter. Therefore, for my familiarity calculation, I have chosen not to apply the 'discounted' aspect of NDCG. The Cumulative Gain (CG), then, of familiarity is the sum of the familiarity score of each song in the playlist. The score does need normalised, however, so that familiarity can be compared across playlists of different length. If the score is simply a count of the familiar songs and artists across a list, longer playlists would perform better by virtue of just containing more songs, and therefore likely more familiar ones.

In the RecSys Competition [29], a correctly recommended song has a relevance score of 1. If the song is not exact, but its artist is correct, this receives a relevance score of 0.25. This is a fair choice when the generation is being evaluated on its ability to predict exact missing tracks. However, this value can be adapted slightly in the case of familiarity, as if a user is familiar with a specific song by an artist, they will likely recognise the artist themselves. However, artists may explore a wide range of musical styles across their discography, and so a user liking a particular song does not necessarily translate into them liking everything by that artist. To balance these two ideas, in my proposed calculation of familiarity, a song with a familiar artist receives a familiarity score of 0.5, a slight increase to the ACM Competition approach. Thus, each song in a playlist can have a familiarity score of: 1, if the song itself is known to the user; 0.5, if the artist is known to the user (but not the specific song); or 0 if neither the song nor artist are known to the user. This more accurately reflects a measure of familiarity, while still maintaining the possibility that it isn't a given that a user will enjoy everything an artist makes. The final familiarity calculation then for one playlist, is given by:

$$Familiarity(F) = \frac{|S_p \cap S_f| + 0.5|A_{(S_p - Sf)} \cap A_f|}{|S_p|}$$

Where Sp is the set of songs in the playlist, Sf is the set of songs familiar to a user, $A_{(S_p - Sf)}$ is the set of artists of the non-familiar songs in the playlist, and Af is the set of artists familiar to a user. The artist familiarity is only evaluated in the case that the specific song is not familiar, since the song already holds a stronger weighting. The final measure has an output within the range [0,1] with 1 representing a playlist that is entirely familiar to the user, and 0 representing a playlist that is entirely new to the user.

The final decision to be made for this calculation is how the sets of familiar songs and artists are to be determined. One approach to this could be to analyse listening

history, which is the idea used for evaluation by [26] as well as in [30]. However, the downside to this approach is that the user may have listened to a track and disliked it. Familiarity could also be calculated on explicit user feedback (such as ratings, or whether a user has 'loved' or 'saved' a song). The issue with this approach is that such data is not readily available. Last.fm [8] for example, does track this information, but it is only accessible for one specific user at a time. For this project, I have chosen to base the set of familiar songs and artists off of users' playlists. This makes the ability to perform a user study evaluation of this measure easier, as participants can simply provide playlists to represent their music library. It is also reasonable to assume users would tend towards adding songs they like to their playlists. This assumption, as mentioned above, cannot necessarily be applied to listening history. Since we are trying to optimise a level of familiarity which will increase user satisfaction, we want the inclusion of 'familiar' songs to benefit the enjoyment of the playlist.

4.2.2 Cohesion

Cohesion, as it applies to playlists, is a measure of how similar the songs contained within a playlist are to each other, and how cohesive the playlist is as a whole. It is referred to in the literature as 'similarity', 'cohesion', and 'homogeneity', which will be used interchangeably here. The idea of playlist cohesion is frequently referred to by participants in user studies as something that affects their opinion of a playlist, and is identified as a key objective measure by Bonnin et al. [25]. Lee's study [44], for example, gave participants 3 playlists based on genres, and asked them to explore why they liked or disliked each. Many of the participants referred to the idea of similarity as part of what formed their opinion on the playlists (*"In fact, a number of participants (P1, P2, P4, P5, P6) reacted positively to the playlists that had coherent set of songs*"). Cunningham et al.'s influential study on playlist creation [33] also references similarity as a desired element for playlist creation, drawing from hundreds of posts and some users of the 'Art of the Mix' playlist discussion forum. The idea of similarity or cohesion also forms the basis of some seminal approaches to APG [46, 20, 21], as well as Fields' [35] comprehensive analysis and subsequent proposal of a novel APG technique. If we refer back to the definition of a playlist from Section 2.2, that a playlist is a collection of songs intended to be listened to together, it is natural that playlists must have some level of cohesion.

Another objective measure that is mentioned by Bonnin et al. [25] and is specifically researched by Sarroff and Casey in [56] is the idea of smooth track transitions between songs in a playlist. Sarroff and Casey use Gaussian Mixture Models to learn how transitions between songs are created in albums, since they are carefully crafted, and then apply this learned meaning to playlists. They achieve good results, their model outperforming their chosen baseline, indicating that the smoothness of transitions does impact playlist quality. A playlist could, of course, have a variety of sounds within it while still being cohesive, and one of the ways it may do this is through these smooth transitions. Hagen [37] concludes as one of the key purposes of playlist creation that a user can curate playlists to have control over how their music is consumed, and one of the ways they do this is through creating an ideal narrative for their listening experience. This smoothness can be accounted for within a cohesion measure, using

pairwise song similarity to calculate how cohesive a playlist is as a whole. A closely related similarity measure is used by Gatzioura et al. which also looks at pairwise song similarity [36] and is used partially in their evaluation.

Cunha, Culdera, and Fujii [32] highlight that there are two main ways song similarity can be calculated: subjectively and objectively. The subjective approach is done by defining similarity based on human descriptions of songs, whereas the objective approach focuses instead on similarity in audio features. In their paper, they use the subjective approach, utilising Last.fm's tagging feature [8]. Last.fm tags are user-defined descriptors for songs, artists, or albums, and are easily accessible via Last.fm's API [9]. Cunha et al. [32] use techniques such as inverse term frequency to define vector similarity between songs based on their user tags. The authors choose to use these tags precisely because they are user-defined, and thus can more accurately capture how humans describe songs. However, in reality, most tags are very closely related to genre descriptions. For example, Gillian Welch's 'Look at Miss Ohio' has the five top tags 'alt-country', 'folk', 'singer-songwriter', 'americana', and 'country' [8]. On the occasion that a song's top tags are not genre descriptors, they do not always give useful information about the song. For example, Bon Iver's 'Rosyln' (from the 'Twilight' film soundtrack) has the top tag 'Twilight', which doesn't necessarily describe the song. Furthermore, the use of genre in playlist research is more frequently referred to in terms of diversity, not similarity, such as by Slaney and White [59], or participants in Lee's user study [44]. In fact, the variety of genres as a quality measure is in part how the variation of a playlist is measured in this proposed metric, which is detailed more in the next section.

Instead, my proposed cohesion measure follows the objective approach, utilising a similarity calculation based on seven audio descriptors taken from Spotify's API [2]. The features include information such as a song's loudness, energy, and danceability. Genre diversity and audio similarity are distinct in my proposed metric, as they are both things that impact the enjoyment of a playlist, and are not necessarily opposites. Songs can both be energetic or danceable, for example, while belonging to different genres or styles, and having these separate measures allows for a playlist to sound cohesive while also maintaining some variety. In addition, in Barrington et al.'s [24] user study on differently generated playlists, 82% of their 185 participants cited similarity between songs' sounds as a key influence on their enjoyment of the given playlists, and was the most cited reason for their opinions. They also find that the participants preferred audio analysis techniques for playlist generation to an approach using tags for similarity.

To calculate the pairwise similarity of two objects, there are different distance measures that can be used. The Cosine Similarity, for example, measures the cosine angle between two vectors to compare how closely related they are. For this metric, the Cosine Similarity measure initially seemed ideal as it outputs in the range [0,1]. However, when tested on playlists, there was not enough variation in the results to allow comparison between different playlists. Two other distance measures used commonly in Machine Learning problems are the Euclidean Distance and the Manhattan Distance. They both take the square root of the sum of differences between each feature in two vectors, however, Euclidean Distance squares this difference, and the Manhattan Distance simply takes its absolute value. Euclidean Distance penalises heavily for large

differences due to the squares and therefore is inappropriate for high dimensions where data may be sparser, which is where Manhattan Distance is more useful. Since in this application only seven dimensions are being used, Euclidean distance is applicable. The strong penalisation of large differences also provides greater variation in the cohesion measure, allowing playlists to be more readily compared. In addition, Euclidean Distance is used by some previous approaches to APG such as [43, 52].

Similarity is, naturally, the opposite of distance. To measure the similarity, we take the distance away from its maximum possible value. Here this would be if all audio values of one vector were 1 and the other were all 0, i.e., the square root of 7. The score is then also normalised by this maximum, mapping the output to the range [0,1]. The similarity between two songs x and y, in this proposal, is then calculated as follows:

$$Sim(x,y) = \frac{\sqrt{7} - \sqrt{(d_x - d_y)^2 + (e_x - e_y)^2 + [...] + (l_x - l_y)^2 + (v_x - v_y)^2}}{\sqrt{7}}$$

Where the vector representations of x and y are made up of elements d = danceability, e = energy, [s = speechiness, a = acousticness, i = instrumentalness], l = liveness, and v = valence, as taken from Spotify's API [2]. The final cohesion metric then takes the pairwise similarity between songs in a playlist, and averages across the length of the playlist minus one. It is calculated on a playlist of length n as follows:

$$Cohesion(C) = \frac{\Sigma_{i=1}^{n-1} Sim(i,i+1)}{n-1}$$

The value produced, in line with the other measures, exists within the range [0,1], 0 being a playlist that is not cohesive at all, and 1 being a playlist made up of songs with entirely the same measures on all acoustic features.

4.2.3 Variety

Variety, also referred to in the literature as diversity or variance, is a measure of how well a playlist contains a mix of songs, which could be based on artist, genre, time period, tempo, or other musical descriptors. Despite being referred to frequently in the discussion about playlist preferences, diversity as a specific measure has not been explored in much detail outwith a few choice papers. A key approach in the area is that of Slaney and White, who sought to calculate the overall diversity of a playlist [59]. They base their measure of diversity on the idea of genre, and create a 'genre space', which playlists can then be mapped to as ellipsoids, and the perceived diversity of the playlist is the volume of the ellipsoid. They explore the diversity of 887 playlists, and in their conclusions *"[they] take [their research] as evidence for users' interest in diverse music."*

Diversity is also a key component in Kamalzadeh's user study [42] of 222 participants. They examine how participants' listening environment impacts their preferences, so the results are reported depending on the level of focus required by the users' in each context. The vast majority (70%) of participants for 'non-attention' activities preferred a mix of 'various moods' throughout their listening session. Even for the 'attention' activities, still 40% preferred variety in their listening, which surprised the authors. In Andric and Haus's user study of 26 participants [22], diversity was also frequently

mentioned as something that affected their enjoyment of playlists, for example *"A good playlist for me is one that contains a lot of diversity"*, and *"my playlists usually contain most diverse songs, just in order not to tire the ear"*.

Variety, especially across genre and artist are mentioned by participants in Lee's study [44], in fact, variance is mentioned as a key indicator of enjoyment by all eight participants. Particularly, for some participants, it was the *lack* of variety, that made them view playlists negatively, e.g. *"It's kind of monotonous. [...] there's no variety."*, and *"three songs by the same artist on the same playlist is kind of, out of five, is a little bit, I think, extreme."* Also in Lee's study, he noted that despite the participants enjoying playlists that they perceived as cohesive, songs that were *too* similar, or playlists with a low variety of artists were viewed negatively. This specifically highlights the need for the balancing of objective measures for user enjoyment.

To calculate the variety of a playlist in my proposed metric, I am using a simple count across songs of unique artists and genres. As discussed in Section 4.2.2 where I lay out the cohesion measure, I have deliberately chosen to keep variety and cohesion distinct from each other, rather than being two sides of the same scale. Artist and genre are specifically chosen for the basis of variety as they are the basis of previous successful work in playlist diversity [59], and referred to by participants in previous user studies on playlist preference and listening habits [42, 44, 22]. Some other options that could be explored for variety would be something like tempo, or time period. However, in Sarroff and Casey's work on song transitions [56], they specifically concluded that tempo didn't have much of an effect on the effectiveness of recommendation, as tempo can be extremely hard to measure accurately, and the datasets with this information are hard to rely on. In terms of time period, Dias et al. [34] identify that in research on 'freshness', release date tended to be uniform across a playlist and that users mostly preferred recent tracks. Thus, using either of these to form a measure of variety would not necessarily be based on user enjoyment.

In my variety measure, genres are obtained using Spotify's API [2] by searching each song's artist. The number of genres returned by the API varies between artists, so a maximum of three top genres are chosen and returned for each artist, and their count stored for normalisation. The simple variety metric is calculated as:

$$Variety(V) = \frac{1}{2}\left(\frac{|A_U|}{|A_P|} + \frac{|G_U|}{|G_r|}\right)$$

Which is the arithmetic mean of the artist and genre counts. A_U is the set of unique artists across a playlist P, and A_P is the set of all artists across P. G_U is the set of unique genres across the playlist P and G_r is the set of retrieved genres across the playlist, capped at a maximum of three per artist. Like the above two measures, the function outputs a number in the range [0,1]

4.2.4 Final Metric

4.2.4.1 User-Specific Ideal Values

As discussed previously, perception of playlist quality and measures like similarity are subjective and depend on how users themselves understand music [33]. Having a

generalised calculation for quality measures is naturally at odds with this subjectivity. Defining similarity, for example, to be based on audio features, only truly represents similarity if a user themselves defines it to be about the way music sounds. However, creating a new measure for each user would be computationally expensive, and also hard to find concrete data for without explicit feedback. What instead seems a reasonable response to this issue is defining user-specific ideal values for each measure. This idea has been loosely applied previously, for example by Pauws and Eggen [52], who take user data to try and understand a user's preferred balance of coherence and diversity and achieve promising results. For example, if a user does perceive similarity differently than based on audio content, an audio-based similarity measure might have a low score on their playlists, and this can then be accounted for in recommendations.

This also applies to specific user behaviours, e.g. regarding familiarity. Maybe some users only wish to create entirely unique playlists, which have no cross-over with the rest of their collection. This clearly differs in behaviour from a user who may create many playlists all utilising a similar overarching set of songs. Generalising ideal familiarity values, in this case, would cause either one of these types of users to be unhappy with the generated playlist. Due to this, the proposed metric relies on user-specific ideal values for each objective measure. These can be calculated based on their existing playlists, and future recommendations are evaluated on how closely they can meet these preferences. For example, the first type of user described above would have a low ideal familiarity value.

4.2.4.2 Combination of Objective Measures

Once a representation is found for the chosen objective measures, the key issue is their combination. The balancing of these measures has been suggested as future work within a number of key papers in the literature, including [25, 35] but is yet to be extensively worked upon. To combine the three chosen objective measures, my proposed metric utilises the Mean Absolute Error (MAE) measure [10].

The optimum way to combine these measures would be to weight their importance based on the extent they impact user enjoyment. These weights could be learned using, for example, linear regression. However, linear regression requires something to be optimised for - which in this case would be user enjoyment. Since there is no publicly available datatset of, e.g., playlist ratings, without human input, these weights could not be accurately learned. Any combination in this project could only be based on estimation. Instead, I have chosen to use an error function that gives each measure equal weighting, but is designed to give a general, overarching impression of how similar two values are: the Mean Absolute Error.

The Mean Absolute Error (MAE) and Mean Squared Error (MSE) are two variations of an error measure commonly used as loss functions in Machine Learning problems. They both average the differences between features in two vectors, but the MSE squares these differences, whereas the MAE takes their absolute value. In MSE, one large error would be penalised more than many small errors, because of the squaring of differences, making it more sensitive to outliers. MAE, instead, gives the same importance to many small errors as it does to one large error. I have chosen to use the

MAE since we cannot necessarily know which objective measures are most important to users, so having differences in all of the measures would be as bad as a large difference in one measure, allowing for a more general calculation of the final metric. The MAE is calculated as follows for a prediction y and a true value x, over n data points:

$$MAE = \frac{\Sigma_{i=1}^{n} |y_i - x_i|}{n}$$

4.2.4.3 Final Metric

Based on the formulations of the three chosen objective measures: familiarity, cohesion, and variety, and the combination technique of the Mean Absolute Error (MAE), I propose the new evaluation metric for playlist quality, the Balanced Objective Measures metric, or *BOM*. For a playlist P, recommended to a user U, the *BOM* score is calculated as:

$$BOM = \frac{1}{3}(|F_P - F_U| + |C_P - C_U| + |V_P - V_U|)$$

Where F_U is the ideal familiarity value for user U, and F_P is the actual familiarity value for playlist P, and so on and so forth for Cohesion C and Variety V. These ideal values (e.g. F_U) are calculated as the arithmetic mean for each measure applied to a user's playlists, and the playlist value is measured just once on the generated list. Each measure, as described in more detail in section 4.2, is calculated as follows on a playlist P, with size n:

$$Familiarity(F) = \frac{|S_P \cap S_f| + 0.5|A_{(S_P - Sf)} \cap A_f|}{|S_P|}$$

$$Cohesion(C) = \frac{\Sigma_{i=1}^{n-1} Sim(i,i+1)}{n-1}$$

Where:

$$Sim(x,y) = \frac{\sqrt{7} - \sqrt{(d_x - d_y)^2 + (e_x - e_y)^2 + [...] + (l_x - l_y)^2 + (v_x - v_y)^2}}{\sqrt{7}}$$

And finally:

$$Variety(V) = \frac{1}{2}\left(\frac{|A_U|}{|A_P|} + \frac{|G_U|}{|G_r|}\right)$$

Since *BOM* is a measure of how closely generated playlists can match user patterns and preferences across the objective measures, a well-performing approach would aim to minimise its score on *BOM*. The output is in the range [0,1], with 0 being a playlist that perfectly matches the user's ideal values.

4.3 Implementation

This section briefly outlines how the final metric was calculated using Python scripts and Spotify's API [2].

4.3.1 Spotify API

The Spotify API for Developers [2] can be used to find in-depth audio analysis for any song hosted on Spotify, as well as information about artists' genres. For the cohesion measure, audio features are extracted about pairwise songs in playlists to form a similarity measure. In the variety measure, a maximum of three genres is returned for each unique artist in the playlist to form part of a measure of diversity across the list. For the user study evaluation, participants provided a set of playlists to be used for the metric calculation. These playlists were migrated to my own Spotify account, and the tool Exportify [5] was used to export the playlists into .csv files, which uses the Spotify ID identifiers for the track, artist, and album titles. These work seamlessly with the API as they can be used to directly retrieve information about tracks or artists.

4.3.2 Implementation

To calculate the metric on each user study participant, Python scripts were written, utilising the 'Spotipy' [17] wrapper for the Spotify API [2]. For the ideal values, each playlist .csv for the participant was loaded into the Python script and the Spotify IDs for each track and its artist were stored as a dictionary. The idealised values were then calculated by looping through each of their playlists and finding the familiarity, cohesion, and variety scores for each. The implementation for the measures was simply the coded version of the formulations presented in Chapter 4. For the generated playlists, the final metric *BOM* was calculated by finding each of the measures for that playlist, and then using the Mean Absolute Error between the ideal user values and the real generated playlist values.

4.4 Use in Practice

Currently, as discussed, the popular method to evaluate APG and APC systems is to compare the recommendations to hand-crafted playlists that already exist within datasets, leaving tracks out and then using Information Retrieval evaluation metrics to measure how well the proposed approaches suggest the missing tracks. The issues with these metrics are discussed in 3.6 in greater detail, but the key problem is that they do not explicitly account for playlist quality. The proposed metric in this research can be used to evaluate *any* playlist and is aimed at being a deliberate measure of playlist quality and user enjoyment.

In practice, the proposed metric could be used in combination with an existing IR metric. They can still be used to evaluate an algorithm's ability to recommend ground-truth playlists, using training and testing sets as is typical in machine learning problems. However, *BOM* would be used in addition to ensure the quality of the recommended playlist, and to account for recommendations that do not necessarily coincide with the ground-truth playlists. A weighted combination of an IR metric and *BOM* could even be found, to understand what balance optimises for user enjoyment. Future APG and APC approaches could then be trained using this new metric.

Chapter 5

User Study Evaluation

To evaluate whether the proposed metric can reasonably capture user enjoyment of playlists, the following user study was designed and undertaken. The study is based on three hypotheses, and has been designed with them in mind. In the following section, the hypotheses, study design decisions, participant information, study protocol, and results are detailed.

5.1 Hypotheses

There are three main hypotheses being tested going into the study, and they have been used in the design decisions for the study itself. These hypotheses are as follows:

1. The majority ($>$50%) of participants will prefer the playlist that performs best on the proposed metric *BOM*.
2. The proposed metric captures user interest more than the benchmark measures of similarity and variety.
3. Participants have more than one objective measure that they take into account when evaluating playlists.

5.2 Benchmarks

To evaluate whether the proposed metric *BOM* can more accurately capture user preference than what currently exists, I have chosen to evaluate it against two chosen benchmark measures.

Ideally, the metric would be benchmarked against one of the IR metrics discussed in Section 3.5. However, since these metrics rely on a set of known relevant documents, i.e. songs in pre-existing playlists, they cannot be used for this user study which recommends entirely new playlists to the participants. Instead, since *BOM* is a measure of playlist quality and user enjoyment, it can be evaluated against other measures of playlist quality that exist in the literature.

To this end, the chosen benchmarks are objective measures proposed in other literature in the APG and APC field. The first of these is the song similarity measure used by Gatzioura et al. [36] in their approach to APC which focuses more specifically on dealing with the semantic gap and better modelling user preference. Since this is a key aim of *BOM*, evaluating against their chosen similarity measure is appropriate. In Pauws and Eggen's influential approach to APG [52], they design their own evaluation technique which also makes use of similarity (and, by their definition, its opposite - diversity). Their research isn't reproducible outwith the environment performed in, as it uses specially created software, but using similarity to evaluate can be done with Gatzioura et al.'s definition in place. The measure is calculated using user-defined tags, and in this project, Last.fm tags [8] are used. The pairwise similarity for each two consecutive songs is computed and then the playlist similarity is the average across the pairs. It is calculated as follows:

$$Similarity = \frac{\Sigma_{i=1}^{n-1} Sim(i,i+1)}{n-1}$$

Where:

$$Sim(a,b) = 1 - log_2(1 + \frac{|a-b|+|b-a|}{|a \cup b|})$$

With a and b representing a list of tags associated with each given song. $|a-b|$ then, is the number of tags associated with a and not b, and vice versa for $|b-a|$. $|a \cup b|$ is the number of tags associated with either a or b.

This similarity measure is used by Gatzioura et al. [36] alongside IR metrics such as precision and recall, as well as also using a diversity measure. Thus, the other benchmark measure is my own variety measure, as defined in Section 4.2.3. Variety, as mentioned previously, is also referred to in both [36] and [52], which both specifically take objective measures into account. Thus, I have chosen to benchmark against both similarity and variety in order to see if *BOM* is capable of outperforming evaluation techniques previously proposed that are close to my own. Since a key aim of the proposed metric *BOM* is specifically to account for multiple objective measures, the benchmarks will also allow us to see if the combination of these measures can more accurately account for user preference than just one on its own. The results for my proposed metric, as well as the two benchmarks, for each playlist given to each participant, is reported in *Table 5.1* in Section 5.8.

5.3 Study Design

The study is targeted at participants who have an interest in music and regularly create and consume playlists so that they can confidently articulate their opinion on the generated playlists, as well as their normal playlist preferences. Participants provide between 10 and 15 of their own or favourite playlists ahead of the study that they feel represent good playlists and have a mix of their musical preferences, which are used to calculate their ideal objective measure values, as well as the familiarity measure for the recommended playlists. They also shared 3 songs that they would like to see a playlist made from. This is done for the participants individually instead of generalised across

participants so that they can talk about an area of music they know well, a technique seen in Lee's study [44] where the participants were able to choose from a selection of genres. Only one of the provided songs is used as a seed song for the study, but due to occasional error with the generation platforms, the others provide alternatives if needed.

Ahead of the study, two different playlists are created based on one of the provided seed songs. One is created using Spotify's 'song radio' feature [16], which creates a playlist based on one song. The other is created using Last.fm's 'create similar tracks playlist' feature [8], which does the same. Only songs that exist on both platforms are chosen, so that the pool of songs is comparable, and the first 15 songs are chosen so that the length of the playlist doesn't affect the participants' opinion. Additionally, the playlists are migrated to the same platform and named 'Test1' and 'Test2', so that there aren't any external influences on the participants' opinion. The participants are asked to listen to the playlists before the session, making sure to consume them independently from each other. The participants are not informed of the difference between the playlists, their scores on the metric, or that a metric has been used to evaluate each. They are simply told they will be interviewed about their enjoyment of the playlists. This ensures impartiality on the participants' side, so any preferences they voice will be based on genuine experience, rather than subconsciously affected by attempting to guess what the playlists are being evaluated on. This approach is also adopted by Pauws and Eggen [52], who choose only to inform participants the study is focusing on features of playlist quality rather than a comparison of differently generated playlists.

During the session, the participants are interviewed in a semi-structured style about the generated playlists and their playlist preferences, as well as on any follow-up points that may arise. After the interview has concluded, the participants are asked to fill in a questionnaire that is more concrete, asking them to directly compare the playlists, and specify what preferences affected their enjoyment. This study design, combining an interview and questionnaire, provides both quantitative and qualitative data. It allows easy comparison of the participants' enjoyment of the provided playlists, which can give a simple conclusion about the success of the proposed metric, but also gives context into why they may like one playlist over the other, and how the proposed metric may have failed to capture their interest.

5.4 Methodology

For the user study, one-to-one, semi-structured interviews combined with post-interview questionnaires were chosen. Having the sessions personalised allows the playlists to be made for each individual, giving them a better knowledge-base to voice their thoughts about the generated playlists. This also allows time for an in-depth exploration of their personal attitudes towards the playlists and what they believe elicits a 'good' playlist. These interviews result in qualitative data of recordings and transcripts, which can be open-coded to find themes and overarching attitudes across the participants.

The post-interview questionnaire provides more quantitative data, which allows for

direct comparison between the playlists for each participant, and gives the ability to statistically analyse the results of the study. The questions are a mix of multiple choice, rating, and opinion-based Likert scales. Asking participants to rate their generated playlists is typical in user studies of this area, such as in [43, 52], and provides a clear and easy measurement of user enjoyment.

5.5 Participants

The participants were chosen because they had an interest in music and regularly consume or create playlists. This is key so that they can confidently express their opinions about the provided playlists. The participants are a mix of people known to the research team and also those who were previously unknown to the research team. The inclusion of the participants known to the research team has the potential to have influenced the results from their interviews, and this is a limitation of the study, which is discussed more in Chapter 6.

Participants were found via advertising to the University's radio station [7], student mailing lists, and through word of mouth of friends of the research team. Factors such as age and gender did not have an affect on participation selection.

5.6 Protocol

Before the study, participants who agreed to join the study were sent a digital copy of the participant information sheet (Appendix A) and the participant consent form (Appendix B) to fill out and return. They were also sent information about how to join a Microsoft Teams [11] session for the scheduled time of their interview.

Participants were asked to provide around 10-15 playlists that they felt were good and represented their music taste, alongside 3 songs that they would like to see playlists based off of, of which one would be picked. In advance of the study, one of the provided seed songs was used to create two different playlists, one using Last.fm [8] and one using Spotify [16]. The user-provided playlists were extracted to .csv files and then used to create the participants' idealised measures for each of familiarity, cohesion, and variety, which form the basis of the evaluation metric. The two generated playlists were evaluated on the proposed metric *BOM*, but this information was not shared with the participant.

The playlists were then given to the participant around 24 hours before the interview was scheduled for, so that they could familiarise themselves with the playlists. Prior to this the playlists were migrated to the same platform and given generic names ('Test 1' and 'Test 2' respectively) to avoid external influence on the participant's opinion. No information about the playlist creation, their performance on the metric, or metadata was known to the participant, simply just the songs. The participants were not instructed to listen to the playlists in a certain way (e.g. on or off shuffle), other than independently of each other so that they would have opinions on each playlist separately. For example, participant P8 provided the seed song 'Stay Useless' by Cloud

Nothings. Their first generated playlist 'Test 1' is as follows:

```
Stay Useless – Cloud Nothings
Hey Cool Kid – Cloud Nothings
The Only Place – Best Coast
Brains – Lower Dens
The House That Heaven Built – Japandroids
Five Seconds – Twin Shadow
Now Here In – Cloud Nothings
Vomit – Girls
The Rat – The Walkmen
Get Away – Yuck
Street Joy – White Denim
No Future / No Past – Cloud Nothings
Demon to Lean On – Wavves
Comeback Kid – Sleigh Bells
Heart in Your Heartbreak – The Pains Of Being Pure At Heart
```

Figure 5.1: P8 'Test 1' Playlist

Which receives the final *BOM* score of 0.1358, with measure values as follows: Familiarity = 0.1667; Similarity = 0.7118; and Variety = 0.7056. The values of similarity, familiarity, and variety respectively across P8's provided playlists can be seen in *Figure 5.2*.

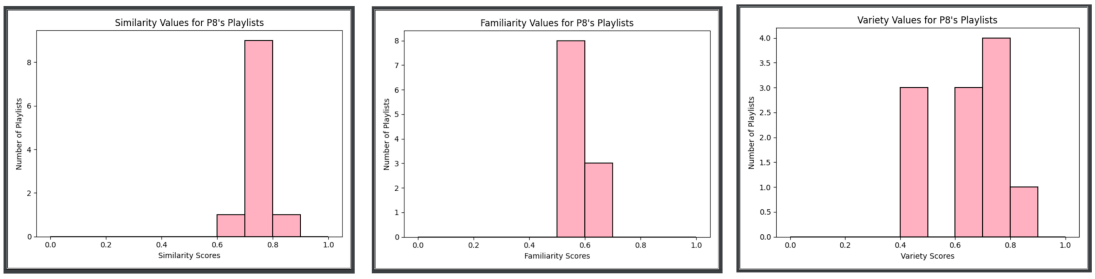

Figure 5.2: P8 Playlist Values

At the start of the interview, participants were reminded of the content of the PIS and Consent Form, and asked to confirm that they were consenting to the session being recorded. The interview then began, and the questions in Section 5.6.1 below were asked in a semi-structured manner, following up with relevant questions based on participant responses. Once the interview concluded, the questionnaire was sent to the participant via Teams chat, and they were given time to complete and send it back. The participant was then asked if they had any questions before the end of the study, which were answered appropriately. The recording was then stopped, the participant thanked for their time, and the session ended.

5.6.1 Interview Questions

1. To start with, do you have any overarching thoughts about the playlists?
2. Did you prefer one playlist over the other? If so, which?

3. If yes, can you articulate why that was?
4. What do you normally look for in a 'good' playlist, whether that is one being made by yourself, or one made by someone/something else?
5. Do you think these playlists, or one of these playlists captured that?

5.6.2 Post-Interview Questionnaire

1. Out of 'Test 1' and 'Test 2', which did you prefer? *(Multiple Choice Question)*
2. On a scale of 1-10 where 1 is 'not at all' and 10 is 'a lot', how much did you enjoy playlist 1? *(Rating scale from 1-10)*
3. On a scale of 1-10, how much did you enjoy playlist 2? *(as above)*
4. My enjoyment of the playlists was affected by prior knowledge of the songs and/or artists *(Likert scale with 7 points ranging from 'Strongly Disagree' to 'Strongly Agree')*
5. My enjoyment of the playlists was affected by the similarity of the songs to each other and to my chosen seed song *(as above)*
6. My enjoyment of the playlists was affected by the mix of different artists and genres *(as above)*
7. My enjoyment of the playlists was affected by something other than the three options above *(as above)*
8. If you answered positively (mostly agree, agree, or strongly agree) to the above question, please let us know what else affected your enjoyment of the playlists *(Long-text answer box)*

5.7 Study Data

In total, nine people participated in the study, in the age range 20-25, made up of five females and four males. The interviews lasted around 10-15 minutes each, totalling 135 minutes of interview footage, and nine questionnaire responses.

5.8 Study Results

In this section, the results of the user study are reported. The results are split into interview and questionnaire. The interview results are based on open-coding analysis, which involves finding common themes across the interview transcripts. These themes, with quotes, are discussed below. The questionnaire data, being more quantitative, has been cross-tabulated. The results in this section are simply reported without much discussion of the findings. This will instead be done in Section 5.8.4.

5.8.1 Metric Results

The scores received on the proposed metric *BOM*, the benchmark similarity calculation, and the benchmark variety calculation for each participant's playlists are shown here, with the better value for each, and the participant's preferred playlist shown in **bold**:

PARTICIPANT	PLAYLIST	BOM	SIMILARITY	VARIETY
P1	**1**	**0.1733**	0.6186	0.6167
P1	2	0.2052	**0.8005**	**0.7167**
P2	1	**0.1732**	0.5949	0.5780
P2	**2**	0.1817	**0.6721**	**0.6495**
P3	1	0.1992	0.7570	**0.7275**
P3	**2**	**0.1768**	**0.7823**	0.6833
P4	1	**0.2053**	**0.8167**	0.8062
P4	**2**	0.2184	0.652	**0.8482**
P5	**1**	0.1314	0.8734	**0.6714**
P5	2	**0.1129**	**0.8862**	0.6225
P6	**1**	**0.1275**	**0.8615**	0.6684
P6	2	0.1964	0.8329	**0.7774**
P7	**1**	**0.1317**	**0.838**	0.6
P7	2	0.1406	0.7243	**0.6333**
P8	1	**0.1358**	**0.7210**	0.7056
P8	**2**	0.1748	0.7034	**0.7471**
P9	**1**	**0.1568**	0.7189	**0.7962**
P9	2	0.1634	**0.7310**	0.7493

Table 5.1: Playlist ratings as given by user study participants

5.8.2 Interview Results

The first three interview questions were specifically related to the provided generated playlists, and the comparison between the two. The final two questions explored what the participants' valued in playlists and whether they felt the generated playlists captured that. All of the participants indicated a preference for one of the two playlists.

While only five of nine participants preferred the playlist that performed better on *BOM*, a common theme across these participants was a stronger preference than there seemed to be in the other four. Participants P1, P3, P6 and P9 all felt positively toward the better performing playlist and negatively toward the worse performing one. P1, for instance, said *"most of the tracks that were in the second playlist, I hadn't listened to before, but I didn't necessarily enjoy them."* P3 also disliked the worse-performing playlist, explicitly referencing a single song - *"I think that's actually probably why I didn't like the playlist cause I really didn't like that song"*. Whereas, participants P2, P4, P5, and P8 who preferred the worse-performing playlist had positive feelings towards both playlists, just with a particular reason for preferring the other. P2, for example, said *"I like the second playlists more, but they were both like, good."* In addition, P5 indicated they really enjoyed both playlists, *"it's really like slight differences [...] in the second one, there was more songs that I loved. But the first one, like, because the flow was better I, I really immersed myself in it"*, and P4 explicitly stated *"I would say that I liked both of them"*.

For their reasonings behind preferring one playlist over the other, participants commonly referred to: their familiarity with the songs or the artists (P1, P2, P3, P5, P6, P7,

P9), for example P3 saying *"test two just had more familiar songs and artists"*; and the overall feeling and similarity of the playlists (P2, P4, P5, P6, P9), such as P6 saying *"I found the first one to be a more kind of relaxing, ambient kind of vibe."* Something that was also common among the participants (P1, P2, P3, P4, P7) was that they enjoyed a mix of familiar and unfamiliar songs, referring to the desire to find new music or new artists by being introduced to them in tandem with things they already enjoyed. For example, from P2: *"I'd sort of want like a bit of a mix of like, things I know. But then like, interspersed with like, new songs that are similar to the ones that I know already"* and P1 *"you sort of find a playlist similar to the stuff that you're currently listening to and then chances are there'll be new tracks on there that you haven't heard before"*. In addition, both P1 and P6 referenced track transitions in their preferred playlists. P6 stated *"there wasn't any, like, aggressive jumps in like, you know, I didn't even realize when one song ended, because it was just like this, like, one long thing."*, and P1 similarly said *"it didn't feel necessarily like it was so jarring when you switch between songs"*.

Some participants had specific curation habits, for example, P8 said they desired *"deep cuts"* in a playlist. For them, this meant they either wanted niche artists they hadn't heard of or less known tracks by familiar artists. Playlist 2, in this participants case, fulfilled these criteria more. Additionally, P5 referenced nostalgia, saying *"sometimes I would put a playlist together that has songs that remind me of a certain time in my life."*. Their preferred playlist, they said, *"had a couple of songs that were what I was listening to a couple of years ago. [...] And it brought me back those memories"*

For participants that preferred the playlist that did not perform best on *BOM* (P2, P4, P5, P8), the most commonly cited reason was that their preferred playlist contained songs that all evoked the same feeling or mood, which was how they organised their own libraries. P5, for example felt that for their preferred playlist *"most of [the songs], with some exceptions [...] were like, sad pop songs, which really, like matched the vibe of 'Ghost of You'"*. P4 also said *"I think more of it is to do with mood, like what kind of mood it puts me in or what kind of feeling the music gives you"*. Another reason given by P8 specifically was that their preferred playlist was more obscure and diverse, *"it's not necessarily very similar, but catches, in a way, the same vibe. So you just need to be in a mood for this, but it's much less obvious and much more interesting."*

5.8.3 Questionnaire Results

5.8.3.1 Questions 1-3

The first three questions deal with the participants thoughts towards the two playlists, and comparison between them. Here, instead of referring to playlists as 'Test 1' or 'Test 2', as they were known to the participants, they will be referred to as either the better-performing and worse-performing playlists, referring to which playlist minimised the score of *BOM*, or as the preferred and non-preferred playlists, referring to whether the playlist was preferred by their respective user. The scores reported across each participant are as follows:

Of the nine participants, five preferred the playlist that performed better on the pro-

Playlist	P1	P2	P3	P4	P5	P6	P7	P8	P9	Mean Average
Better-Peforming	9	6	8	8	6	9	8	6	10	7.78
Worse-Peforming	4	8	5	10	9	5	6	8	5	6.67

Table 5.2: Playlist ratings as given by user study participants

posed metric *BOM*. Of these five participants, the average score given to their preferred playlist out of 10 was 8.8, and the average score given to their non-preferred playlist was 5.0. For the four participants that preferred the worse performing playlist, the average score given to their preferred playlist was 8.75, and the average score given to their non-preferred playlist was 6.5.

5.8.3.2 Questions 4-9

Questions four to nine are more subjective questions, asking the participants to indicate what affected their enjoyment of the provided playlists. Seven of nine participants (all except P4 and P8) agreed that their enjoyment of the playlists was affected by their prior knowledge of the songs or artists contained, three participants (P1, P2, and P6) strongly agreeing. Seven of nine participants (except P3 and P8) also agreed that their enjoyment of the playlists was affected by the similarity of the songs to each other and to their chosen seed song, four participants (P2, P4, P5, P6) strongly agreeing. All participants except P2 agreed that their enjoyment of the playlists was affected by the mix of artists and genres, two (P1, P8) strongly agreeing.

Enjoyment of playlist was affected by:	Agreement	Neutral	Disagreement
Familiarity of songs / artists	78%	11%	11%
Similarity of songs	78%	11%	11%
Mix of artists / genres	89%	11%	0%
Something else	33%	56%	11%

Table 5.3: Agreement levels among participants for each statement

In addition, three participants (P3, P5, P8) indicated that something else affected their enjoyment of the playlists that was not covered by questions 4-8, and explained further in question 9. P3 detailed *"Second playlist was more upbeat, happier vibes. Overall better, liked most songs as opposed to a few bad songs ruining my opinion of the first playlist"*. P5 said *"the mood associated with listening to these songs"*, and similarly P8 indicated *"Mood or overall feeling of songs. Something that is hard to put into any objective measures"*.

5.8.4 Results Discussion

From the study results, hypothesis 1) has proved to be true, however not by a large margin, and not enough to be statistically significant. The majority of participants *did* prefer the playlist that performed better on *BOM*, but this majority was very slim at only 55%. Hypothesis 2) has not been proven, with song similarity and variety, on a quantitative level, each being able to predict enjoyment for the same level of participants at also 55%. Hypotheses 3) was proved to be true for all participants, as

they all indicated through interview and questionnaire responses that there were at least two objective measures they considered when thinking about playlist quality.

The results indicate that while not perfect in its current form, *BOM* does seem to be a reasonable starting point for the creation of a metric like this in the future. 100% of participants indicated more than one objective measure affected their enjoyment of the playlists, which is important to consider in future work in this area and solidifies Bonnin et al. [25], and Fields' [35] attitudes that their combination is key for future research. Also, a majority (78%) of participants indicated that their familiarity with the playlists impacted their opinion. Familiarity is looked at least across previous responses to playlist evaluation, for example, novel evaluation techniques used in [36, 52] focus on song similarity and some kind of representation of diversity. Neither consider familiarity, yet the results of this study indicate that it is important to listeners, at least across these participants.

Also interesting is the level of preference the participants tended to have when looking at the playlists that performed well on *BOM*. None of the participants gave a better-performing playlist a score of less than six, and all indicated mostly positive attitudes towards these playlists, usually just having a specific reason that the other playlist was better. The playlists, across all participants, that performed better on *BOM*, were on average rated higher, with a mean score of 7.78, compared to the worse-performing playlists average rating of 6.67. While not a very large margin, this could at least indicate that playlists performing well on *BOM* are generally enjoyed, and with future work into weighting or further personalisation of the metric for each user, it could be a good predictor of user enjoyment.

The benchmark measures also indicate where issues with *BOM* may lie and how it could be improved in future research. As mentioned previously, and discussed further in the next chapter, *BOM* does not weight different objective measures, instead giving them all the same importance. However, we can see from the benchmark scores that participants tended to prefer the playlist with either higher similarity or higher variety. If *BOM* gave more weight to whichever of these the participant preferred, it may even better indicate user enjoyment. The benchmarks themselves also were unable to entirely predict user enjoyment, with both also predicting five of nine participants' preference. While *BOM* does not outperform these benchmarks, it does indicate that for all participants, more affected their enjoyment than just one measure of similarity or variety. The basis for *BOM* is the combination of these objective measures and understanding how they can impact users' satisfaction with generated playlists. The results of this study do show that just one objective measure cannot generally predict user preference well enough to be used as the basis for an evaluation metric.

Finally, one of the most interesting findings from this user study that should continue to be considered in research in this area, is that the semantic gap [27, 36] does tend to be the main issue in being unable to model user preference. As discussed in Section 5.8.2, in the cases participants preferred the playlist that performed worse on *BOM*, three of the four referenced the mood of the songs or the feelings the playlist evoked in them, which is of course, harder to represent using computation, as we cannot necessarily understand what feelings are being evoked for a user, or why this is the case.

Chapter 6

Discussion

6.1 Limitations

Since this project was undertaken on a relatively small scale, there are limitations to the research presented in the chapters above. This mainly falls into two categories: limitations of the proposed metric itself, and limitations of its evaluation.

6.1.1 Metric Limitations

In terms of the proposed metric itself, there are a few limitations to its creation and potential use in the future. Firstly, as mentioned in some sections above, each measure combined into the metric has the same impact on the final score, i.e. they each hold the same weight. This limits the opportunity for the metric to be truly personalised for each user. It can be seen clearly in the results of the user study of the metric against the benchmarks in *Table 5.1* that most participants valued one of the benchmark measures over the other. From the interview and questionnaire results it was also apparent that the participants had their own specific ideas of what constituted a good playlist, and while basing the metric on personalised ideal values goes some of the way in capturing this, it could be one of the reasons the metric didn't perform as well as expected.

Additionally, the formulations for the measures used in the metric are arguably quite simplistic. This is especially relevant for the similarity measure, and there have been a number of influential works on calculating song similarity, e.g. in [60, 48, 49, 47]. These approaches typically perform their own spectral analysis, producing features they identify as important, rather than using predefined features like in Spotify's API [2], which we have no control over. Using a pre-calculated audio analysis limits what features we can base the similarity measure on, and those used may not accurately capture how users truly perceive similarity.

6.1.2 User Study Limitations

The user study also has limitations in terms of how conclusive the results from it can be. Most importantly, the user study was very limited in terms of its size. With only

nine participants, findings cannot be generalised across all playlist consumers, and the results of the study were not enough to be statistically significant. The user study also contains limitations in that some of the participants knew the research team prior to the study taking place. While none of the participants knew the real motivation behind the study or had any knowledge about the contents of this research, there is still the potential for bias from these known participants.

6.2 Future Work

There is lots of opportunity for future work to expand on the base idea of my proposed metric. Again, this falls into two main categories: improvements to the metric itself, and a more expansive evaluation of the metric's effectiveness.

6.2.1 Metric Future Work

Currently, the calculation of the metric relies on relatively simple formulations for each objective measure. For the cohesion measure for example, there are other explored approaches in music recommendation literature on how to calculate similarity between songs. Conducting more research into which audio features impact user enjoyment and implementing one of the more computationally complex similarity measures within *BOM* could improve its ability to capture user preference. Similarly for the familiarity measure, more user playlists, their entire song library, or their listening history could be used as a more accurate representation of songs and artists the user knows. Like discussed in Section 4.2.1., there are some key issues here such as making sure the user also *likes* the songs they've previously listened to, but in a more resourceful exploration of this metric, there may be ways to combat this.

Another key improvement to the metric would be implementing weights in some way in the final calculation, either using a weighted version of the MAE, or something else such as a weighted harmonic mean. It may be that more closely meeting variety values have a greater impact on user enjoyment, for example, than cohesion. In which case, the metric could more strongly penalise a recommended playlist that was further off the user's ideal variety value. Like discussed in Section 4.2.4.2, choosing these weights is a challenge in itself, since it is hard to optimise for wide-ranging user enjoyment without readily available playlist ratings, or a very large user study. If in future, those issues could be combated with the public release of a playlist rating dataset, or with improvements in user simulation, these weights could make an improvement to *BOM*.

Finally, a particularly interesting application for the use of a metric such as the one proposed here would be the use specifically in APC. Since we know from previous research into playlist preferences [33] that the playlist purpose is the key to understanding what it should contain, for APC, we need to understand this purpose in order to make better recommendations. In an APC scenario, the metric could be specifically applied to the pre-existing playlist for the 'ideal' values, and recommendations should try and keep the balance of those values as close to the original tracklisting as possible.

6.2.2 User Study Future Work

As discussed in Section 6.1.2, there are two main limitations to the user study performed here, those being that the study was limited in size, and also that there is the potential for bias as some of the participants were known to the research team prior to the study. While the metric proved to be reasonable for this particular group, the study was not large enough to be statistically significant, and the results cannot be generalised to all playlist creators. A wider ranging user study would further determine whether the use of multiple objective measures can capture user interest. Particularly something that may be useful would be A/B Testing on two groups that receive, discuss, and rate playlists generated by two different models, one trained using a current evaluation technique such as missing tracks and IR metrics, and one trained using a combination of this and the metric proposed here.

6.3 Research Question

Throughout this project, the research question I have aimed to answer is as follows:

Can 'objective measures' of playlist quality be combined to create an evaluation metric that captures user preference for automatic playlist recommendations?

6.4 Conclusions

To conclude, in this project, I have aimed to propose and test a new evaluation metric to be used in Automatic Playlist Generation and Automatic Playlist Continuation as a measure of playlist recommendation quality. I have researched and analysed the current techniques used in APG and APC research, and discussed their issues in terms of how they reflect user preference, to deepen my understanding of the problem and find what needs to be changed. The proposed metric, Balanced Objective Measures (*BOM*), combines user familiarity, playlist cohesion, and song variety, to reflect user preferences for playlists as discussed throughout previous literature and aims to solve the discussed problems with existing evaluation techniques.

When evaluated on a small user study, while not perfect, *BOM* does show a good starting point for future work in the creation of a playlist evaluation metric, and indicates that combining objective measures *can*, on some level, capture user preference for playlist recommendations. Playlists performing well on *BOM* had a high average score across participants when compared to worse-performing playlists, indicating that it does capture a level of user enjoyment. In addition, participants in the study further show the need for the combination of multiple objective measures in future playlist quality metrics, as benchmark values of similarity and variety could also not perfectly predict user preference, and all participants indicated more than one measure that was important to them in playlist curation. In the cases that *BOM*'s performance could not predict user preference, participants' responses solidified that music is a subjective recommendation item that relies heavily on users' moods and emotions, and further work on music recommendation must have this at its core.

Bibliography

[1] Collaborative filtering. https://developers.google.com/machine-learning/recommendation/collaborative/basics.

[2] Documentation: Spotify for developers. https://developer.spotify.com/documentation/, journal=Documentation — Spotify for Developers.

[3] The echo nest. https://www.crunchbase.com/organization/the-echo-nest.

[4] Evaluate your recommendation engine using ndcg. https://towardsdatascience.com/evaluate-your-recommendation-engine-using-ndcg-759a851452d1.

[5] Exportify. https://github.com/pavelkomarov/exportify.

[6] F-score. https://deepai.org/machine-learning-glossary-and-terms/f-score.

[7] FreshAir Radio. https://freshair.radio/.

[8] Last.fm. https://last.fm.

[9] Last.fm API. https://www.last.fm/api.

[10] MAE and RMSE: which metric is better? https://medium.com/human-in-a-machine-world/mae-and-rmse-which-metric-is-better-e60ac3bde13d.

[11] Microsoft teams. https://www.microsoft.com/en-gb/microsoft-teams/group-chat-software.

[12] MRR vs MAP vs NDCG: Rank-aware evaluation metrics and when to use them. https://medium.com/swlh/rank-aware-recsys-evaluation-metrics-5191bba16832.

[13] Nielsen music U.S. music 360 report highlights. https://www.nielsen.com/wp-content/uploads/sites/3/2019/04/us-music-360-highlights.pdf.

[14] Playlists overtake albums listenership. https://musicbiz.org/news/playlists-overtake-albums-listenership-says-loop-study/.

[15] Precision and recall. https://developers.google.com/machine-learning/crash-course/classification/precision-and-recall.

[16] Spotify. https://spotify.com.

[17] Spotipy. https://spotipy.readthedocs.io/en/2.17.1/.

[18] State of the streaming nation 2 report. https://www.midiaresearch.com/blog/announcing-midias-state-of-the-streaming-nation-2-report.

[19] Spotify company info. https://newsroom.spotify.com/company-info/, Mar 2021.

[20] M. Alghoniemy and A. Tewfik. A network flow model for playlist generation. In *2001 IEEE International Conference on Multimedia and Expo*, pages 329,330,331,332, 2001.

[21] M. Alghoniemy and A. H. Tewfik. Personalized music distribution. In *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.00CH37100)*, volume 4, pages 2433–2436 vol.4, 2000.

[22] Andreja Andric and Goffredo Haus. Estimating quality of playlists by sight. In *First International Conference on Automated Production of Cross Media Content for Multi-Channel Distribution (AXMEDIS'05)*, pages 7–pp. IEEE, 2005.

[23] J-J Aucouturier and François Pachet. Scaling up music playlist generation. In *Proceedings. IEEE International Conference on Multimedia and Expo*, volume 1, pages 105–108. IEEE, 2002.

[24] Luke Barrington, Reid Oda, and Gert RG Lanckriet. Smarter than genius? human evaluation of music recommender systems. In *ISMIR*, volume 9, pages 357–362. Citeseer, 2009.

[25] Geoffray Bonnin and Dietmar Jannach. Automated generation of music playlists: Survey and experiments. *ACM Computing Surveys (CSUR)*, 47(2):1–35, 2014.

[26] Klaas Bosteels, Elias Pampalk, and Etienne E Kerre. Evaluating and analysing dynamic playlist generation heuristics using radio logs and fuzzy set theory. In *ISMIR*, volume 9, pages 351–356, 2009.

[27] Oscar Celma, Perfecto Herrera, and Xavier Serra. Bridging the music semantic gap. 01 2005.

[28] Chih-Ming Chen, Chun-Yao Yang, Chih-Chun Hsia, Yian Chen, and Ming-Feng Tsai. Music playlist recommendation via preference embedding. In *RecSys Posters*, 2016.

[29] Ching-Wei Chen, Paul Lamere, Markus Schedl, and Hamed Zamani. Recsys challenge 2018: Automatic music playlist continuation. New York, NY, USA, 2018. Association for Computing Machinery.

[30] Chung-Yi Chi, Richard Tzong-Han Tsai, Jeng-You Lai, and Jane Yung-jen Hsu. A reinforcement learning approach to emotion-based automatic playlist gener-

ation. In *2010 International Conference on Technologies and Applications of Artificial Intelligence*, pages 60–65. IEEE, 2010.

[31] Nick Craswell. *R-Precision*, pages 2453–2453. Springer US, Boston, MA, 2009.

[32] Renato LF Cunha, Evandro Caldeira, and Luciana Fujii. Determining song similarity via machine learning techniques and tagging information. *arXiv preprint arXiv:1704.03844*, 2017.

[33] Sally Jo Cunningham, David Bainbridge, and Annette Falconer. " more of an art than a science": Supporting the creation of playlists and mixes. 2006.

[34] Ricardo Dias, Daniel Gonçalves, and Manuel J Fonseca. From manual to assisted playlist creation: a survey. *Multimedia Tools and Applications*, 76(12):14375–14403, 2017.

[35] Benjamin Fields et al. *Contextualize your listening: The playlist as recommendation engine*. PhD thesis, Goldsmiths College (University of London), 2011.

[36] A. Gatzioura, J. Vinagre, A. M. Jorge, and M. Sànchez-Marrè. A hybrid recommender system for improving automatic playlist continuation. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1, 2019.

[37] Anja Nylund Hagen. The playlist experience: Personal playlists in music streaming services. *Popular Music and Society*, 38(5):625–645, 2015.

[38] Derek L Hansen and Jennifer Golbeck. Mixing it up: recommending collections of items. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1217–1226, 2009.

[39] Balázs Hidasi, Alexandros Karatzoglou, L. Baltrunas, and D. Tikk. Session-based recommendations with recurrent neural networks. *CoRR*, abs/1511.06939, 2016.

[40] Rosilde Tatiana Irene, Clara Borrelli, Massimiliano Zanoni, Michele Buccoli, and Augusto Sarti. Automatic playlist generation using convolutional neural networks and recurrent neural networks. In *2019 27th European Signal Processing Conference (EUSIPCO)*, pages 1–5. IEEE, 2019.

[41] Dietmar Jannach, Iman Kamehkhosh, and Geoffray Bonnin. Biases in automated music playlist generation: A comparison of next-track recommending techniques. In *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization*, pages 281–285, 2016.

[42] Mohsen Kamalzadeh, Dominikus Baur, and Torsten Möller. A survey on music listening and management behaviours. 2012.

[43] Peter Knees, Tim Pohle, Markus Schedl, and Gerhard Widmer. Combining audio-based similarity with web-based data to accelerate automatic music playlist generation. In *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, pages 147–154, 2006.

[44] Jin Ha Lee, Bobby Bare, and Gary Meek. How similar is too similar?: Exploring users' perceptions of similarity in playlist evaluation. In *ISMIR*, volume 11, pages 109–114, 2011.

[45] Elad Liebman, Maytal Saar-Tsechansky, and Peter Stone. Dj-mc: A reinforcement-learning agent for music playlist recommendation. *arXiv preprint arXiv:1401.1880*, 2014.

[46] Beth Logan. Content-based playlist generation: Exploratory experiments. In *ISMIR*, volume 2, pages 295–296, 2002.

[47] Beth Logan and Ariel Salomon. A music similarity function based on signal analysis. In *ICME*, pages 22–25, 2001.

[48] François Maillet, Douglas Eck, Guillaume Desjardins, Paul Lamere, et al. Steerable playlist generation by learning song similarity from radio station playlists. In *ISMIR*, pages 345–350, 2009.

[49] Matija Marolt. A mid-level melody-based representation for calculating audio similarity. In *ISMIR*, pages 280–285, 2006.

[50] Brian McFee and Gert RG Lanckriet. The natural language of playlists. In *ISMIR*, volume 11, pages 537–541, 2011.

[51] Francois Pachet, Pierre Roy, and D. Cazaly. A combinatorial approach to content-based music selection. 1:457–462 vol.1, 07 1999.

[52] Steffen Pauws and Berry Eggen. Pats: Realization and user evaluation of an automatic playlist generator. In *ISMIR*, volume 2, pages 222–230, 2002.

[53] Martin Pichl, Eva Zangerle, and Günther Specht. Towards a context-aware music recommendation approach: What is hidden in the playlist name? In *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pages 1360–1365. IEEE, 2015.

[54] John C Platt, Christopher JC Burges, Steven Swenson, Christopher Weare, and Alice Zheng. Learning a gaussian process prior for automatically generating music playlists. In *NIPS*, pages 1425–1432, 2001.

[55] David MW Powers. What the f-measure doesn't measure: Features, flaws, fallacies and fixes. *arXiv preprint arXiv:1503.06410*, 2015.

[56] Andy M Sarroff and Michael Casey. Modeling and predicting song adjacencies in commercial albums. *Proc. SMC*, 2012.

[57] Zamani-H. Chen C. et al. Schedl, M. Current challenges and visions in music recommender systems research. 2018.

[58] Leslie F. Sikos. *The Semantic Gap*, pages 51–66. Springer International Publishing, Cham, 2017.

[59] Malcolm Slaney and William White. Measuring playlist diversity for recommendation systems. In *Proceedings of the 1st ACM workshop on Audio and music computing multimedia*, pages 77–82, 2006.

[60] Ofer Tchernichovski, Fernando Nottebohm, Ching Elizabeth Ho, Bijan Pesaran, and Partha Pratim Mitra. A procedure for an automated measurement of song similarity. *Animal behaviour*, 59(6):1167–1176, 2000.

[61] Thanh Tran, Renee Sweeney, and Kyumin Lee. Adversarial mahalanobis distance-based attentive song recommender for automatic playlist continuation. *CoRR*, abs/1906.03450, 2019.

[62] A. Vall, M. Dorfer, Hamid Eghbal-zadeh, M. Schedl, Keki Burjorjee, and G. Widmer. Feature-combination hybrid recommender systems for automated music playlist continuation. *User Modeling and User-Adapted Interaction*, 29:527–572, 2018.

[63] Andreu Vall, Massimo Quadrana, Markus Schedl, and Gerhard Widmer. The importance of song context and song order in automated music playlist generation, 07 2018.

[64] Maksims Volkovs, Himanshu Rai, Zhaoyue Cheng, Ga Wu, Yichao Lu, and Scott Sanner. Two-stage model for automatic playlist continuation at scale. New York, NY, USA, 2018. Association for Computing Machinery.

[65] Morgan K Ward, Joseph K Goodman, and Julie R Irwin. The same old song: The power of familiarity in music choice. *Marketing Letters*, 25(1):1–11, 2014.

[66] Hamed Zamani, Markus Schedl, Paul Lamere, and Ching-Wei Chen. An analysis of approaches taken in the acm recsys challenge 2018 for automatic music playlist continuation. 10(5), 2019.

Appendix A

Participant Information Sheet

Participant Information Sheet

Project title:	Evaluation of Automatically Generated Playlists
Principal investigator:	Pavlos Andreadis
Researcher collecting data:	Jennifer Logan
Funder (if applicable):	N/A

This study was certified according to the Informatics Research Ethics Process, RT number 2019/33469. Please take time to read the following information carefully. You should keep this page for your records.

Who are the researchers?

The research is being carried out as part of an honours project looking at how we evaluate playlists that have been generated automatically. Today's study is run by Jennifer Logan who is the student undertaking the project, with help from supervisor Pavlos Andreadis.

What is the purpose of the study?

The purpose of this study is to help understand what makes users like or dislike playlists and how the idea of a 'good' playlist should be defined and evaluated.

Why have I been asked to take part?

We are looking for people who have an interest in music and create their own playlists. You have been asked to participate because we believe that you have this type of experience.

Do I have to take part?

No – participation in this study is entirely up to you. You can withdraw from the study at any time, up until 12th April 2021 without giving a reason. After this point, personal data will be deleted and anonymised data will be combined such that it is impossible to remove individual information from the analysis. Your rights will not be affected. If you wish to withdraw, contact the PI. We will keep copies of your original consent, and of your withdrawal request.

What will happen if I decide to take part?

You will be asked to provide us with some of your own playlists and some 'seed songs' which a new playlist can be based on. During the session you will be shown some automatically generated playlists made specially for you and will be allowed to listen to a snippet of the songs. If you consent, the session will be recorded so we

THE UNIVERSITY of EDINBURGH
informatics

can review it in more detail later. After you have familiarised yourself with the generated playlists, you will be asked for your opinion on each of them. Our goal is to understand what makes playlists agreeable to users, so hearing your opinion on differently generated ones will help us understand what approaches to automatically generating playlists should take into account. The session should take around 30-45 minutes.

Are there any risks associated with taking part?

There are no significant risks associated with participation.

Are there any benefits associated with taking part?

None other than helping a student with their project and potentially understanding your own playlist preferences more!

What will happen to the results of this study?

The results of this study may be summarised in published articles, reports and presentations. Quotes or key findings will be anonymised: We will remove any information that could, in our assessment, allow anyone to identify you. With your consent, information can also be used for future research. Your data may be archived for a maximum of 4 years. All potentially identifiable data will be deleted within this timeframe if it has not already been deleted as part of anonymization.

Data protection and confidentiality.

Your data will be processed in accordance with Data Protection Law. All information collected about you will be kept strictly confidential. Your data will be referred to by a unique participant number rather than by name. Your data will only be viewed by the researcher/research team Jennifer Logan and Pavlos Andreadis.

All electronic data will be stored on a password-protected encrypted computer, on the School of Informatics' secure file servers, or on the University's secure encrypted cloud storage services (DataShare, ownCloud, or Sharepoint) and all paper records will be stored in a locked filing cabinet in the PI's office. Your consent information will be kept separately from your responses in order to minimise risk.

What are my data protection rights?

The University of Edinburgh is a Data Controller for the information you provide. You have the right to access information held about you. Your right of access can be exercised in accordance Data Protection Law. You also have other rights including rights of correction, erasure and objection. For more details, including the right to lodge a complaint with the

THE UNIVERSITY of EDINBURGH
informatics

Information Commissioner's Office, please visit www.ico.org.uk. Questions, comments and requests about your personal data can also be sent to the University Data Protection Officer at dpo@ed.ac.uk.

Who can I contact?

If you have any further questions about the study, please contact the lead researcher, Jennifer Logan <s1704329@ed.ac.uk>.

If you wish to make a complaint about the study, please contact inf-ethics@inf.ed.ac.uk. When you contact us, please provide the study title and detail the nature of your complaint.

Updated information.

If the research project changes in any way, an updated Participant Information Sheet will be made available on http://web.inf.ed.ac.uk/infweb/research/study-updates.

Alternative formats.

To request this document in an alternative format, such as large print or on coloured paper, please contact Jennifer Logan <s1704329@ed.ac.uk>.

General information.

For general information about how we use your data, go to: edin.ac/privacy-research

Appendix B

Participant Consent Form

Participant number:_____

Participant Consent Form

Project title:	Evaluation of Automatically Generated Playlists
Principal investigator (PI):	Pavlos Andreadis
Researcher:	Jennifer Logan
PI contact details:	pavlos.andreadis@ed.ac.uk

By participating in the study you agree that:

- I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.
- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

1.	I agree to being audio recorded.	**Yes**	**No**
2.	I agree to being video recorded.	**Yes**	**No**
3.	I allow my data to be used in future ethically approved research.	**Yes**	**No**
4.	I agree to take part in this study.	**Yes**	**No**

Name of person giving consent	Date dd/mm/yy	Signature

Name of person taking consent	Date	Signature

THE UNIVERSITY of EDINBURGH informatics