UNIVERSITY COLLEGE LONDON

MSc THESIS

---

# Leveraging Audio States & Transitions for Improved Track Sequencing in Music Streaming Sessions

---

*Author:*
Aaron NG

*Supervisors:*
Rishabh MEHROTRA, Spotify
Emine YILMAZ, UCL

*This report is submitted as part requirement for the MSc Degree in Machine Learning at University College London. It is substantially the result of my own work except where explicitly indicated in the text. The report may be freely copied and distributed provided the source is explicitly acknowledged.*

September 11, 2020

# *Abstract*

The sequential nature of music streaming, specifically information on track sequences, play a pivotal role in the quality of music recommendation. In this thesis, we focus on two distinct parts: (i) we investigate the role that audio characteristics of music content play in music streaming sessions and (ii) leverage those insights on audio characteristics to improve track sequencing in sessions. In the first part, we formulate transitions in audio attributes in a session as a changepoint detection problem, and extract latent states of different audio attributes within each session. Based on analysis of large scale music streaming sessions from a popular music streaming platform, we explore the extent to which audio characteristics fluctuate within streaming sessions and their impact on user satisfaction. We highlight the promise of leveraging the extracted audio states and transitions in better sequencing of tracks by demonstrating the potential gains in user satisfaction on offer. Next, we utilise the insights on audio states and transitions to propose a three-step track ranking model. The model first determines the top audio attribute that best characterises the session, followed by predicting the preferred audio state, and use both information to sequence tracks in the session. We show that our ranking model indeed improves key user satisfaction metrics in streaming sessions compared to various baselines, while not overexposing popular content.

# *Acknowledgements*

First and foremost, I would like to thank my supervisor Rishabh Mehrotra, who helped me immensely in shaping this thesis and without whom this project would not have gotten very far. Not only did he guide me on the research process, he emphasised the importance of distilling complex research ideas to succinct and clear stories. The acceptance and publication of my first paper in an academic conference as a result of this thesis would not have been possible without his encouragement and collaboration.

Next, I would like to thank various people from Spotify, including Mounia Lalmas, Brian Brost, Samuel Way, Simon Durand, Marco Marchini, for attending my presentation based on the research done in this thesis and providing thoughtful comments and feedback. Also, I wouldn't have the chance to work with Rishabh if not for the recommendation by Emine Yilmaz. Lastly, I want to thank all my friends and family for their support all these years.

# Contents

# 1 Introduction

## 1.1 Motivation

The increasing popularity of online music streaming platforms, each serving tens of millions of users [40], has created the need to develop automated tools that can algorithmically generate personalised music experiences. These experiences should be cohesive and engaging. Not only should the music recommender system recommend music that matches the current listening context and preferences of the user, the tracks should be sequenced in such a way that "*flows smoothly*" from one song to the next. Sequencing tracks has been regarded as "*more of an art than a science*" [11], with the ordering of tracks playing an important role in the quality and cohesiveness of the playlist.

Audio characteristics (e.g. rhythm, harmony, tempo) help to describe and quantify acoustic properties of music content. These characteristics complement information more commonly associated with playlists, namely organisational logic (i.e. tracks belong to a particular performing artist or musical genre) and usage information (i.e. listening behaviour and user interactions) [5]. For example, transitioning between a smooth classical piano piece and a high tempo dance track will likely create a sudden and uncomfortable change in the listening experience.

Analysing the fluctuations of these properties across listening sessions

can help us understand its impact on user perception of recommended music. Ultimately, the notion of flow and its attainment is an aesthetic phenomenon; a user may want the tempo to stay relatively constant or neighboring songs to be acoustically similar as a function of creative intent. If songs are to be cross-faded, proper sequencing can help ensure that consecutive pairs of songs have similar keys and tempos, allowing for less abrupt transitions.

Hence, in this thesis, we draw inspiration from the importance of track sequencing in enhancing user experience within music streaming sessions. We focus on learning how audio attributes affect the listening experience and leverage the representations we learnt to develop track re-ranking algorithms that can optimise user satisfaction of recommended music.

## 1.2 Research Outline

The thesis focuses on understanding and extracting audio states & transitions from music streaming sessions, followed by leveraging the extracted information to better sequence tracks for improved user satisfaction. Thus, we can formulate two research themes:

In the first part, we begin by considering audio attributes in streaming sessions and attempt to quantify how these audio attributes vary in streaming sessions. We propose to formulate this problem as a changepoint detection task performed on sequences of audio attribute values in a session, where each sequence of audio attribute value is treated as an independent time series. Using a public dataset of over 150 million music streaming sessions, we perform a large scale analysis of audio states and transitions, using Hidden Markov Models to model the latent states of the sequences. Specifically, we ask these questions: 1) How varied are audio properties of tracks within a session, and how common are those fluctuations? 2) How do these

variations differ across the different audio attributes? 3) Are variations in audio properties related to user satisfaction? 4) Can insights about audio states & transitions help in track sequencing for improved user satisfaction? Most importantly, we run a counterfactual track re-ranking experiment that shows that the extracted audio states and transitions can indeed be leveraged to improve user satisfaction in streaming sessions, which motivates the second part of the research.

In the second part, we motivate the need to develop a track re-ranking model that utilises audio states and transitions to better sequence tracks. We define the issues we need to address whilst developing this model. We propose a novel three-step ranking model that: 1) attempts to predict the most suitable audio attributes to use to re-rank each session, 2) determine the preferred states in the selected audio attributes and 3) define a re-ranking logic using the audio attributes and preferred states. Subsequently, we evaluate the performance of our model and its variants with that of other baselines. We also analyse our results across different types of sessions to see if our model performs better or worse on particular subsets of sessions.

## 1.3 Thesis Outline

Chapter 2 provides an overview of Music Information Retrieval (MIR), detailing various approaches that have traditionally been used to quantify characteristics of audio tracks and methods to automatically sequence tracks for optimal listening experiences. Chapter 3 encapsulates the first part of our research. We start by defining research questions we would like to address, and explore the dataset that will be used. Next, we cover our novel method for extracting audio states from tracks in streaming sessions, and present our

analysis on its impact on user satisfaction, which motivates the need for development of track sequencing algorithms that utilise audio attributes to improve listening experiences. The second part of our research is described in Chapter 4. In Chapter 4, we propose a ranking model that leverages our findings, and show that our model can indeed help to improve key user satisfaction metrics. We also present experimental results on several variants of the model and analyse the model's performance across different subsets of sessions. Lastly, we conclude the thesis in Chapter 5, where we summarise the findings and implications, and provide recommendations on future work.

## 1.4 Origins

A portion of the thesis, in particular the methods and findings of the first part of our research (Chapter 3), have been submitted and accepted as a publication at the Fourteenth ACM Conference on Recommender Systems (RecSys '20): *Investigating the Impact of Audio States & Transitions for Track Sequencing in Music Streaming Sessions* [43] by Ng and Mehrotra. Ng performed the experiments while all authors contributed to the text.

## 1.5 Code Repository

Code that has been written to build the models, run the experiments and analyse the results in this thesis can be found in:

`https://github.com/aaronng91/msc-thesis-audio-states`

# 2 Background and Related Work

In this chapter, we provide an overview to information retrieval in music and review relevant works that involve extracting acoustic characteristics of audio tracks. We also review track sequencing algorithms that have traditionally been used to automatically generate listening experiences for users. Lastly, a background context to latent variable models employed in this thesis is provided.

## 2.1 Music Information Retrieval

Traditional ways of listening and discovering music, such as through radio broadcasts and record stores, are rapidly being replaced by personalised ways of hearing and learning about music [7]. With the immense growth of online music streaming platforms in recent years (e.g. Spotify [1], Pandora [2], Youtube Music [3]), which individually have millions of tracks in their music collections, this has posed a major challenge to the indexing and retrieval of music. This spurs the fast growing field of Music Information Retrieval (MIR), which is primarily concerned with the extraction and inference of meaningful features from music, and the development of different search and retrieval schemes [13].

Using contextual metadata, which can be roughly divided into factual

[1] https://www.spotify.com/
[2] https://www.pandora.com/
[3] https://music.youtube.com/

(i.e. artist, duration, album) and subjective (i.e. genre, mood, style) metadata, have long been a common approach to MIR and has enjoyed wide popularity [7, 51]. However, there are problems with relying only on metadata, since metadata are annotated by humans which are prone to error. Subjective metadata also represent opinions of those annotating them, which makes good editorial supervision particularly important [15]. These problems are magnified when we consider the curation of millions of tracks. This is where content-based approaches come into play, by providing an unbiased, quantitative alternative that can complement context-based methods.

TABLE 2.1: Task lists for MIREX 2005/06. Adapted from [36].

| **2005** | **2006** |
|---|---|
| Audio Artist Identification | Audio Beat Tracking |
| Audio Drum Detection | Audio Cover Song Identification |
| Audio Genre Identification | Audio Melody Extraction |
| Audio Melody Extraction | Audio Music Similarity and Retrieval |
| Audio Onset Detection | Audio Onset Detection |
| Audio Tempo Extraction | Audio Tempo Extraction |
| Audio Key Finding | Query-by-Singing or Humming |
| Symbolic Genre Classification | Score Following |
| Symbolic Key Finding | Symbolic Melodic Similarity |
| Symbolic Melodic Similarity | |

Content-based MIR consider musical concepts such as the rhythmic, timbre or harmonic characteristics of audio tracks. However, quantifying and extracting musical concepts from audio signals is not an easy task, therefore being the subject of intensive research. Some of the research areas being actively addressed in content-based MIR include identification of the presence of vocal regions, categorisation of songs based on emotional patterns of both vocal and non-vocal segments, recognition of performing artist/mood/genre, and many more [42]. In this regard, the Music Information Retrieval Evaluation eXchange (MIREX) [36] was set up in 2005 to provide the necessary infrastructure, tasks, datasets and evaluation framework for MIR researchers. A list of the tasks used in MIREX 2005 and 2006 is shown in Table 2.1, which

showcases the wide variety of standardised tasks and evaluation that have been identified in the field. MIREX has since fostered great advancements in MIR [14] and can be used as an indicator for the state-of-the-art in the field.

## 2.2 Audio Feature Extraction

Audio features can be mainly classified into low-level, mid-level and high-level features. *Low-level* features are extracted from short audio segments of length 10-100ms, such as timbre or temporal features [42]. *Mid-level* features are extracted from words, syllables, notes or a combination of low-level features, such as pitch, harmony and rhythm [29]. Lastly, *high-level* features label the entire track and provide semantic information. Commonly known features such as genre, instrument, mood fall into this category. Likewise, the techniques being used to extract audio features also vary across the different levels of features, which we will describe below.

### 2.2.1 Methods

In general, low-level features are normally extracted using signal processing techniques. Firstly, audio signals are transformed using transformation methods like Discrete Cosine Transform [2], Fast Fourier Transform [45] or constant-Q transform [50]. From the spectrum obtained, spectral features such as Mel-Frequency Cepstral Coefficients, spectral flatness measures, amplitude spectrum envelope can be extracted [10, 27, 33]. Besides the adoption of features commonly associated with signal processing as described above, statistical methods are also used to capture temporal variations into audio signals. Parameters like mean, variance, kurtosis, and a combination of them [53, 37] can be used to form feature vectors. Probabilistic models such as Hidden Markov Models (HMM) have also been used to extract temporal features [48], of which the underlying concepts will be explored later in this chapter.

Mid-level features are normally derived from more specific algorithms, such as pitch values being extracted using frequency estimation and pitch analysis algorithms [52, 30]. Harmony, of which chord sequences play a major role, can be extracted by a variety of chord-detection algorithms [25, 41]. Rhythmic features such as beats per minute or tempo can be computed by the recurrence of the most repeated pattern in an audio track, or the envelope of an auto-correlation of the audio signal [42]. However, better results in MIR tasks can often be obtained by combining low and mid-level features [29]. Given the combinatorial explosion of features, feature selection also becomes paramount when selecting the ideal set of features for MIR tasks [3, 17].

Lastly, high-level features, which are usually categorical features, are extracted from low and mid-level features using a variety of classification models. Supervised classification models such as k-nearest neighbours (KNN), support vector machines (SVM), Gaussian mixture models (GMM) and artificial neural networks (ANN) have frequently been used. For example, [20] has used KNN on Wavelet Packet representation of audio signals for genre classification. [23] have also proposed using a SVM on intensity, pitch, timbre, tonality and rhythm to perform genre classification. In the identification of vocal sections, [18] have applied a two-state HMM with vocal and non-vocal states on melody information. It is noteworthy to mention that there is no single classification model that works the best for any specific high-level feature, since the performance of different classification models vary with the type of low to mid-level features being used.

## 2.3 Track Sequencing Approaches

The idea of playlist generation, or sequencing tracks in a way that fulfills a target characteristic in the best possible way, can be considered as an intersection between the fields of MIR and Recommender Systems (RS). Music

streaming platforms and web radio services strongly rely on the automation of playlist generation in order to serve highly personalised playlists to millions of users. However, large scale automated track sequencing comes with its challenges. Not only do we need high quality metadata and extracted audio characteristics which was discussed in the previous section, we also need to identify the user's intent and context correctly in order to recommend tracks that best suit their taste [5]. Specifically, track recommendation approaches should be able to dynamically adapt its recommendations to meet the user's current intent, as tracks are often immediately consumed [32]. An example is the incorporation of immediate user feedback such as 'skip' or 'like' actions from users [47].

Besides track metadata and audio characteristics, online music platforms also have the advantage of being able to easily track the usage and listening behaviour of their users. This include logs of listening sessions, the popularity of individual tracks, interactions with other users, which all serve as useful inputs to the track recommendation approaches [56].

Next, we will review various algorithmic approaches to track sequencing, which can be organised into three categories: Similarity-based, Collaborative Filtering and other approaches.

### 2.3.1 Similarity-based Approaches

A straightforward strategy to track recommendation is to select and order tracks based on their similarities. One would need to start by using available data to come up with some kind of track representation. Then, one would need to choose a distance function in order to measure similarity between tracks, such as Euclidean distance [31] or Kullback-Leibler (KL) divergence [54]. Subsequently, tracks can be selected and sequenced based on similarity to a seed track [34], to previously selected tracks [9] or to tracks that user have

historically liked [46]. Many works have adopted a multi-pronged approach by combining track metadata, user usage history and track co-occurrence patterns before applying a similarity measure [24, 21, 38].

However, one shortfall of similarity-based approaches is that homogeneity of the tracks is often the main or only criteria. This approach may therefore not work well when users are in the mood for discovering new and diverse content, rather than consuming similar songs repeatedly.

### 2.3.2 Collaborative Filtering

The usage of Collaborative Filtering (CF) in the field of RS has been widely popular since their effectiveness in movie recommendations have been demonstrated in [4]. CF is the process of filtering or evaluating items through the opinions of other people [49], which can either be memory-based or model-based. Memory-based approaches recommend tracks consumed by other users with similar preferences to the target user, or recommend tracks similar to tracks that the target user has already consumed before. On the other hand, model-based approaches model latent representations of the users and tracks directly, through methods such as matrix factorisation or neural networks. Alternative ways of applying CF were also explored in [21], where playlists are considered as users, and tracks appearing in similar playlists are considered as potential tracks for inclusion in the generated playlist.

Despite its popularity, CF suffers from the "cold-start problem", which happens when there is not much information about new users, or when rating information is sparse. The system will therefore not have enough information to make suitable recommendations to new users. This has led to works involving a hybrid approach to recommendation, by incorporating CF with other classification algorithms and similarity techniques [35, 8].

### 2.3.3 Other Approaches

Besides the more commonly used similarity-based and CF approaches, many works have explored alternative approaches to track sequencing. Instead of measuring similarity, pattern mining approaches attempt to identify patterns in the data using association rules and sequential patterns [1]. For example, [21] propose a model based on mining sequential patterns of latent topics based on track tags. Statistical approaches using HMMs or Latent Dirichlet Allocation (LDA) have also been well explored in [39, 22]. Lastly, hybrid methods combining two or more approaches mentioned in this section are commonly used to combine advantages of different techniques while avoiding drawbacks of individual techniques [39, 21].

## 2.4 Latent Variable Models

Part of the thesis builds upon the idea of extracting underlying representations of audio characteristics in streaming sessions. Latent variable models are statistical models that relate observed variables that are directly measured to hidden variables that are inferred rather than directly observed. In particular, we focus on the usage of Hidden Markov Models in this thesis, a prominent choice of latent variable model for time series data.

### 2.4.1 Hidden Markov Models

#### Overview

A Hidden Markov Model (HMM) is a tool for representing probability distributions over sequences of observations [19], depicted graphically in Figure 2.1. It is defined by two main properties. First, the observation at time $t$, denoted by $\mathbf{x}_t$ is generated by some hidden process, whose states are discrete and denoted by $s_t \in \{1 \dots K\}$. Second, if we consider a first-order HMM, the

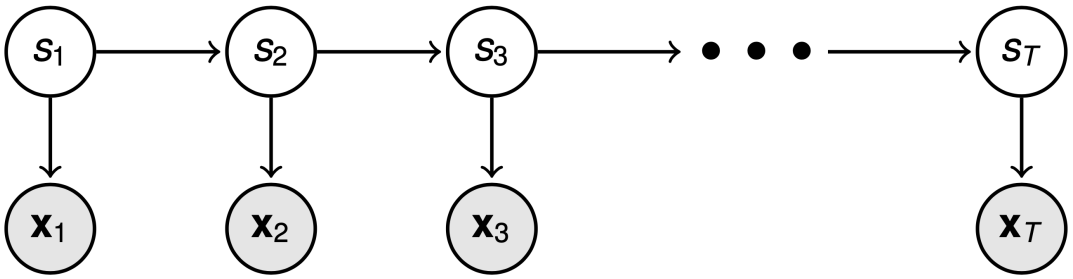



FIGURE 2.1: A graphical representation of HMM, where $s_t$ are the hidden states and $\mathbf{x}_t$ are the observations.

hidden process satisfies the *Markov property*, which states that the probability of the current state $s_t$ is independent of all the states prior to $t-1$, given the previous state $s_{t-1}$:

$$P(s_t|S_1 \dots s_{t-1}) = P(s_t|s_{t-1}) \tag{2.1}$$

The probability of an output observation $\mathbf{x}_t$ also satisfies the Markov property: it depends only on $s_t$, not on any other states or observations. Taking these two properties, the joint distribution of a sequence of states and observations can be defined as:

$$P(s_{1:T}, \mathbf{x}_{1:T}) = P(s_1)P(\mathbf{x}_1|s_1)\prod_{t=2}^{T} P(s_t|s_{t-1})P(\mathbf{x}_t|s_t) \tag{2.2}$$

Next, to fully specify a HMM, we would need to define the following components:

- Initial probability distribution over the hidden states: $\pi_i = P(s_1 = i)$, which represent the probability that the HMM will start in state $i$.
- State transition matrix representing the probability of moving from state $i$ to state $j$ for all pairs of states: $\Phi_{ij} = P(s_t = j|s_{t-1} = i)$
- Emission probability distribution over the hidden states: $A_i(\mathbf{x}) = P(\mathbf{x}_t = \mathbf{x}_t|s_t = i)$, which represent the probability of an observation $\mathbf{x}_t$ being generated from state $i$.

### Training

In order to learn the parameters of a HMM, namely the transition probabilities $\Phi$ and emission probabilities $A$, the standard algorithm used is the Expectation Maximisation (EM) algorithm. EM is an iterative algorithm that works by computing an initial estimate for the probabilities, then using those estimates to compute a better estimate, and this process is repeated until convergence. More specifically, if we define the free energy $\mathcal{F}$ of the system as:

$$\mathcal{F}(q, \theta) = \sum_{s_{1:T}} q(s_{1:T})(\log P(\mathbf{x}_{1:T}, s_{1:T}|\theta) - \log q(s_{1:T})) \tag{2.3}$$

where $\theta$ is the collection of parameters $\{\Phi, A, \Pi\}$ and $q(s_t) = p(s_t|\mathbf{x}_t, \theta)$, the expectation step (E-step) can be defined as the maximisation of $\mathcal{F}$ with respect to $q$ with $\theta$ fixed:

$$q^* = \text{argmax}_q \, \mathcal{F}(q, \theta) \tag{2.4}$$

and the maximisation step (M-step) can be defined as the maximisation of $\mathcal{F}$ with respect to $\theta$ with $q$ fixed:

$$\theta^* = \text{argmax}_\theta \, \mathcal{F}(q^*, \theta). \tag{2.5}$$

Both steps are repeated one after another until a suitable convergence criteria is met, giving us the learned parameters of the HMM.

### Inference

To infer the most likely state which the model believes the system was in at each timestep $t$, we compute the marginal probability distribution $P(s_t|\mathbf{x}_{1:T})$. Given a trained HMM and a sequence of observations, this likelihood computation step can be efficiently done using the Forward algorithm, a form of

dynamic programming. If we define $\alpha_t(i) = P(\mathbf{x}_{1:T}, s_t = i|\theta)$, $\alpha_t(i)$ can be computed iteratively by summing over the extensions of all the paths that lead to state $i$, in the form:

$$\alpha_t(i) = \sum_{i=1}^{K} \alpha_{t-1}(i) P(s_t|s_{t-1}) P(\mathbf{x}_t|s_t) \tag{2.6}$$

Finally, the marginal probability distribution at state $i$ can be obtained via:

$$P(s_t = i|\mathbf{x}_{1:T}, \theta) = \frac{\alpha_t(i)}{\sum_k \alpha_t(k)} \tag{2.7}$$

However, knowing the most likely state at each timestep is not the same as the most probable sequence of states inferred from the data. To compute the single best path, we would need to use the Viterbi decoding algorithm. This is very similar to the Forward algorithm, where the only difference is that the maximum over all possible previous paths are taken instead of the sum. Thus, if we define $v_t(i) = P(\mathbf{x}_{1:T}, s_{1:t}|\theta)$, $v_t(i)$ can be computed iteratively in the form:

$$v_t(i) = \text{argmax}_k \, v_{t-1}(i) P(s_t|s_{t-1}) P(\mathbf{x}_t|s_t) \tag{2.8}$$

# 3 Extracting and Understanding Audio States & Transitions

In this chapter, we discuss the first part of our research. We start by listing key research questions to be addressed, describe the dataset being used, propose a state extraction model, and lastly, present our findings.

## 3.1 Research Questions

We hypothesise that understanding how audio characteristics of tracks vary within streaming sessions, and their impact on user satisfaction can help us improve models for track sequencing. To investigate our hypothesis, we formulate and propose four research questions.

RQ1 **How varied are audio properties of tracks within a session?** *Do audio properties change at all across tracks in streaming sessions? How common are these audio fluctuations?*

RQ2 **How do variations in audio characteristics differ across the different audio attributes?** *Are these variations in audio states across different audio attributes heterogeneous? How about transitions between audio states?*

RQ3 **Are variations in audio properties related to user satisfaction?** *Are state transitions correlated with skip behaviour? How do users respond to such fluctuations? Are the audio states correlated with user satisfaction?*

RQ4 **Can insights about audio states & transitions help in track sequencing for improved user satisfaction?** *How can we use information about audio states & transitions to perform track re-ranking?*

## 3.2 Data Context

To investigate the research questions, we use the Music Streaming Sessions Dataset (MSSD) [6], a large scale public collection of listening sessions and associated user actions on Spotify. MSSD contains approximately 150 million listening sessions gathered over a period of about 2 months. Each streaming session typically has 10 or more tracks, and a cut off is set at 20 tracks per session. Each track of each session contain user actions such as skips, pauses, forwards and listening context. Track metadata (e.g. US popularity estimate, duration, release year) are also provided, together with 18 unique audio attributes, listed in Table 3.1.

TABLE 3.1: List of 18 audio features in each track

| | | | |
|---|---|---|---|
| acousticness | beat_strength | bounciness | danceability |
| dyn_range_mean | energy | flatness | instrumentalness |
| key | liveness | loudness | mechanism |
| mode | organism | speechiness | tempo |
| | time_signature | valence | |

We begin by exploring two main features we will be using in our experiments, session length and skips. As there are many variations of skip behaviour being provided in MSSD, we use 'skip_2' (boolean indicating if the track was only played briefly) as our measurement for skip behaviour. The left plot of Figure 3.1 shows that session length decreases steadily with increasing length, with the exception of session lengths of 20, which is the by-product of longer sessions being capped at this length. If we measure the average skip rate across different session lengths, we also observe that skip rates are higher for longer sessions (right plot of Figure 3.1).

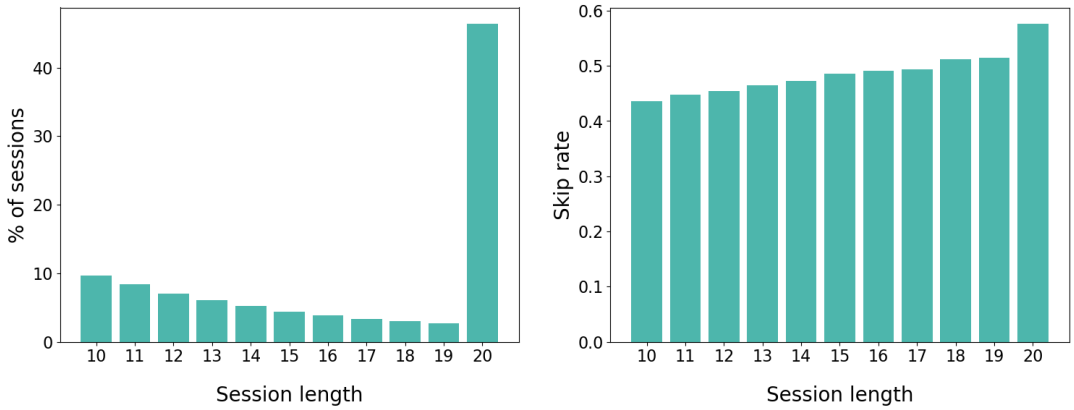



FIGURE 3.1: Left: Distribution of session lengths. Right: Session length against skip rate

The audio attributes are also of particular interest to us in this piece of research. Examples of these attributes include: 1) Acousticness (a confidence measure of whether the track is acoustic, 2) Energy (a perceptual measure of intensity and activity in the track), 3) Liveness (Likelihood of the presence of an audience in the recording). The full list and definitions of all audio features can be found in the Spotify API [1].

By analysing the distributions of some of the audio attribute values in Figure 3.2, we notice that the distributions vary across different features. While most features have standard distributions, few features have heavily skewed or even bimodal distributions. Examples of these features include 'flatness', 'instrumentalness', 'dyn_range_mean'. Such skewed distribution of values are potentially going to impact results when we analyse variations in audio features later on. Moreover, by computing the Pearson correlation between audio attributes, as shown in Figure 3.3, we noticed that there exists small or no correlation between most features. However, there are some exceptions such as 'beat_strength', 'bounciness' and 'danceability', which we can consider merging them later on.

[1] https://developer.spotify.com/documentation/web-api/reference/tracks/get-several-audio-features/

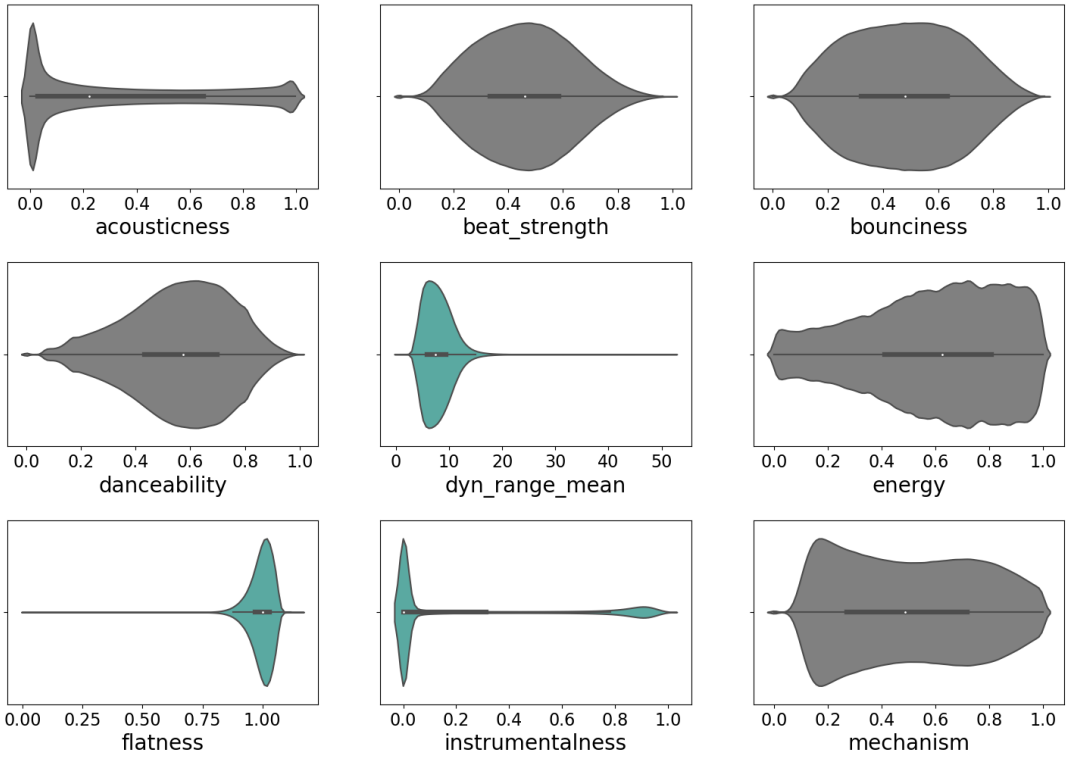



FIGURE 3.2: Distribution of values for selected audio attributes

For our experiments, we select a random sample of 50,000 listening sessions from MSSD. Sessions that are in shuffle mode, or containing different listening contexts, are filtered out, so that we only analyse sessions in a single context where tracks are already sequenced using a predefined measure. Lastly, we define a state or session as satisfying (SAT) when the average number of skips $\leq 0.25$, and dissatisfying (DSAT) when the average number of skips $\geq 0.75$.

## 3.3 Proposed State Extraction Model

We propose a state extraction model using changepoint detection to find underlying representations of the variations in audio features in listening sessions. In each session, we treat each sequence of audio feature values across the tracks as a time series. Since we have 18 audio attributes per session, this means 18 independent time series per session.

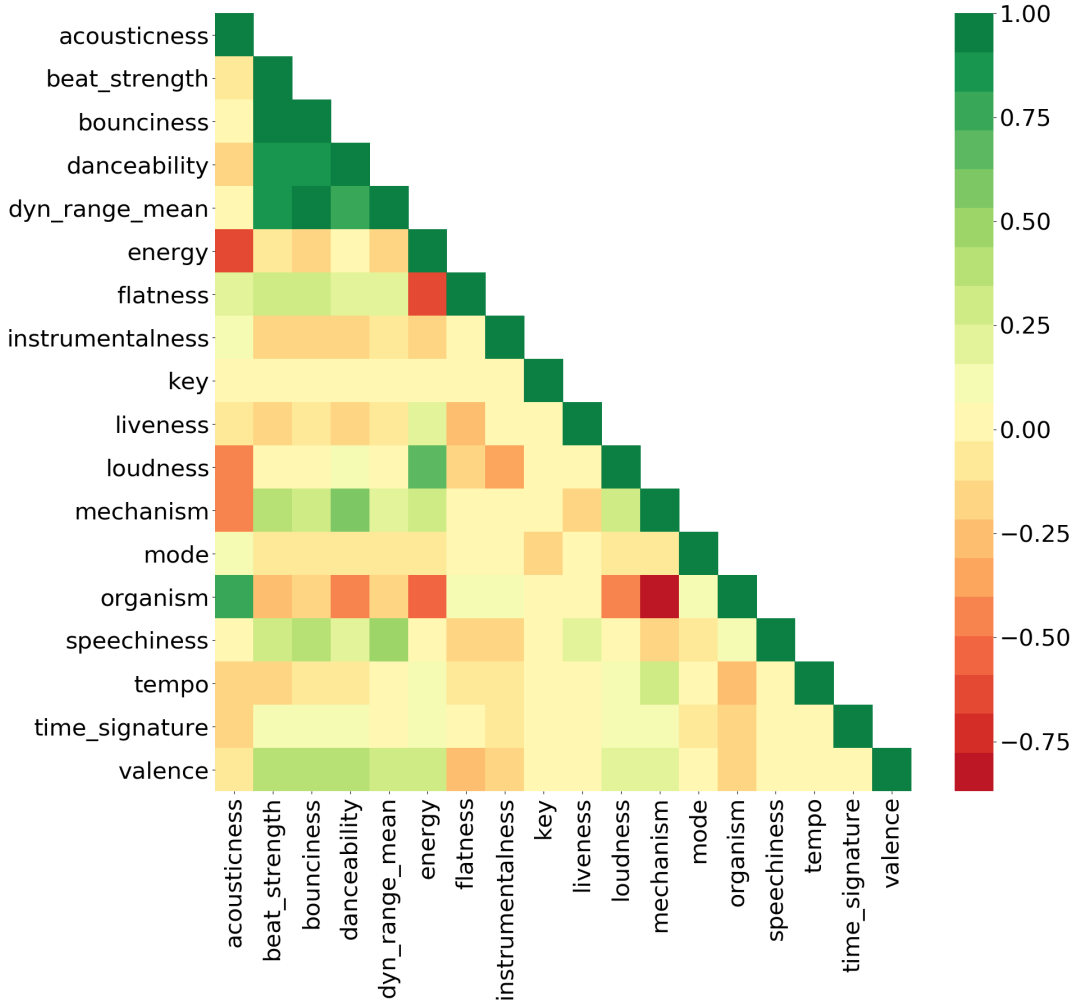



FIGURE 3.3: Pearson Correlation matrix between all 18 audio attributes

The latent states of each time series are modelled using a HMM. In our model, we define $k$ discrete latent states $s_t \in \{1, 2, \ldots, k\}$. A uniform categorical distribution is used for the state transition probabilities, such that the probability of staying in the previous state or transiting to another state is uniform, which gives us:

$$s_t | s_{t-1} \sim \text{Cat}(\{\frac{1}{k}, \ldots, \frac{1}{k}\}) \tag{3.1}$$

The emission probabilities are defined using a normal distribution:

$$\mathbf{x}_t \sim \mathcal{N}(\mu_{z_t}, \sigma^2_{\text{feat}}) \tag{3.2}$$

where $\mu_{s_t}$ is the mean of the trainable latent states, and $\sigma^2_{\text{feat}}$ is the average standard deviation of the corresponding audio attribute across all sessions. A prior on the latent states $s_t \sim \mathcal{N}(\mu_{\text{feat}}, \sigma^2_{\text{feat}})$ is also defined, using the average mean and standard deviation of the corresponding audio attribute across all sessions.

We run an Adam [28] optimiser with a learning rate of 0.1 to train the model and compute the Maximum a Posteriori (MAP) fit to the observed values:

$$\boldsymbol{\mu}_{MAP} = \text{argmax}_{\boldsymbol{\mu}}\ p(s_{1:T}|x_{1:T}) \tag{3.3}$$

Once the model is trained, we compute the marginal posterior distribution $p(S_t = i|x_{1:T})$ over the states for each track and assign the most likely state to each track:

$$s_t^* = \text{argmax}_{s_t}\ p(s_t|x_{1:T}) \tag{3.4}$$

$k$ was set to 10 in our experiments, but latent states with similar means are combined post-inference. States with fewer than 3 tracks are treated as outliers and removed, since states with only 1-2 tracks are not meaningful. As each session has a maximum of 20 tracks and each state should be meaningful (i.e. consisting of 3 or more tracks), we are not going to exceed 10 latent states, which is why $k$ was set to 10.

In Figure 3.4, we present the original values and the extracted states for the 'danceability' audio attribute in a sample session. The model has determined that there are 3 underlying states in the 'danceability' of this session, where tracks 1, 2, 4, 7... belong to 1 state, tracks 8, 9, 16, 17 ... belonging to another state, and the rest of the tracks belonging to a third state.

Our proposed model considers both global and local variations of audio

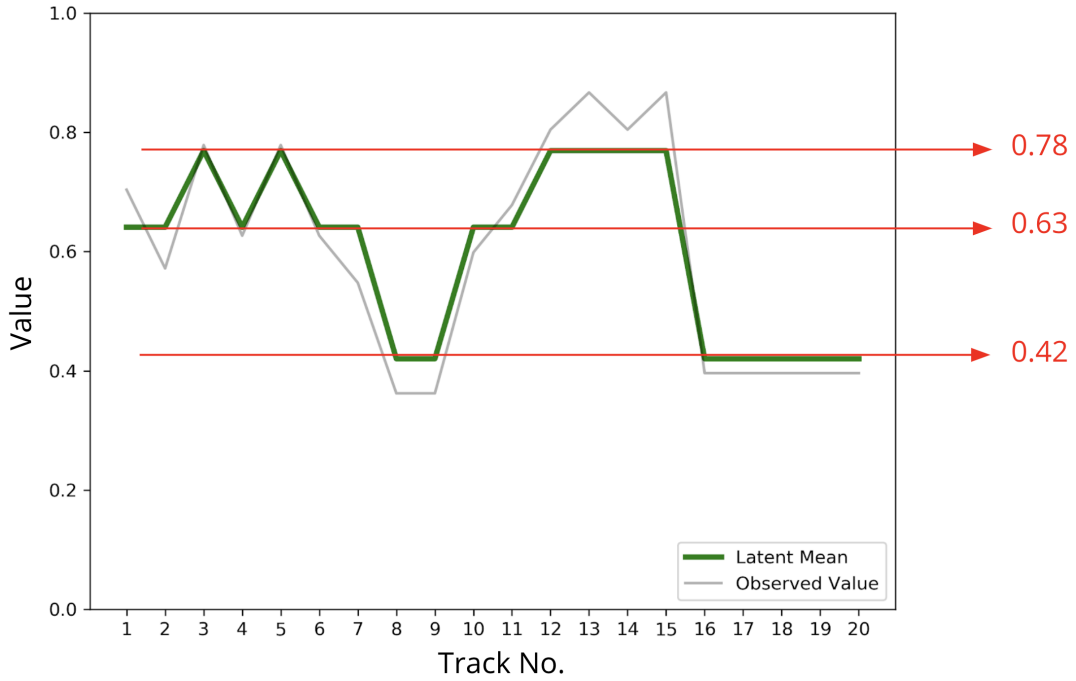



FIGURE 3.4: Vector of 'danceability' values in a sample session of 20 tracks. Green line represents the latent mean assigned to each track through our state extraction model. A total of 3 latent means (0.42, 0.63, 0.78) were found in this example.

characteristics in listening sessions. Local variations are captured as we fit a HMM per audio attribute per session. This is important since user preferences differ across different sessions. Yet, global variations are also taken into account since parameters of our proposed model are calculated from global distributions of the audio attribute values.

## 3.4 Findings

### 3.4.1 Audio properties of tracks vary within a session

By analysing the states extracted for every feature in every listening session through the proposed model, we observe that audio attributes do change across tracks in listening sessions. Over 96% of the sessions have at least 1 audio attribute found to have some variation (i.e 2 or more states) across the session, as seen Figure 3.5. Yet, the distribution peaks at 4 features per

session, which indicates that most sessions are characterised by only a few features, rather than all of them.

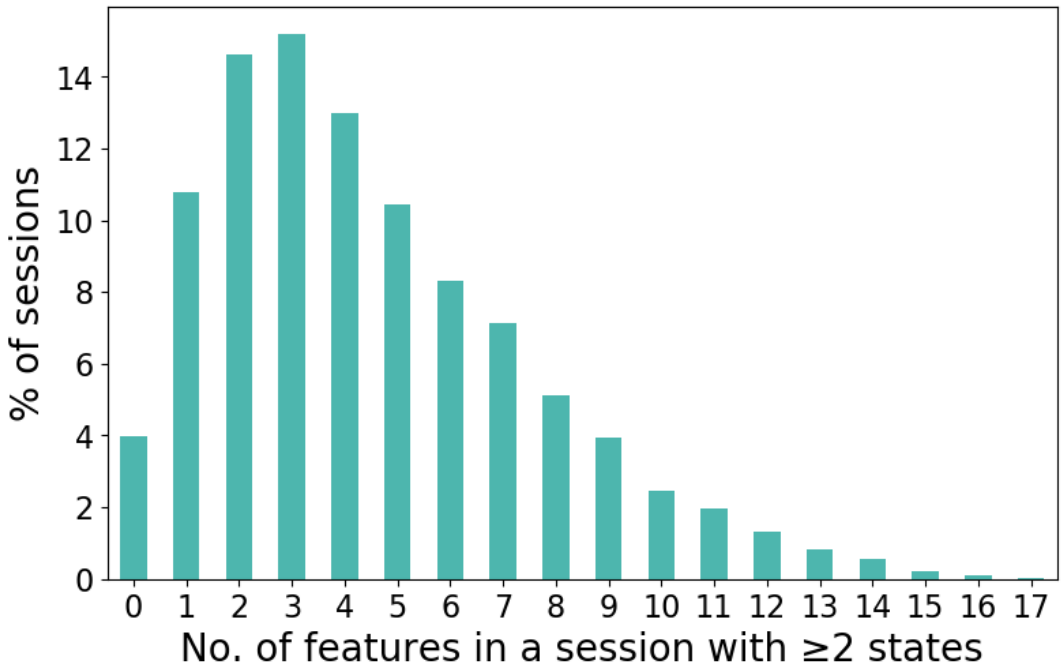



FIGURE 3.5: Overall distribution of sessions with N features that have 2 or more states

Fluctuations in audio attribute values are also fairly prevalent. If we consider 4 sample listening sessions and their extracted states for 2 audio attributes, as shown in Figure 3.6, we can observe that the number of states vary across different features in different sessions. Some features exhibit no variations (Loudness in Session 3), many features have 2 states (Features in Sessions 2 and 4), while some features even have 3 states (Danceability in Session 1). The number of transitions between states also vary, with some features transitioning only once (Features in Sessions 2 and 4), while some features exhibit many transitions (Features in Session 1).

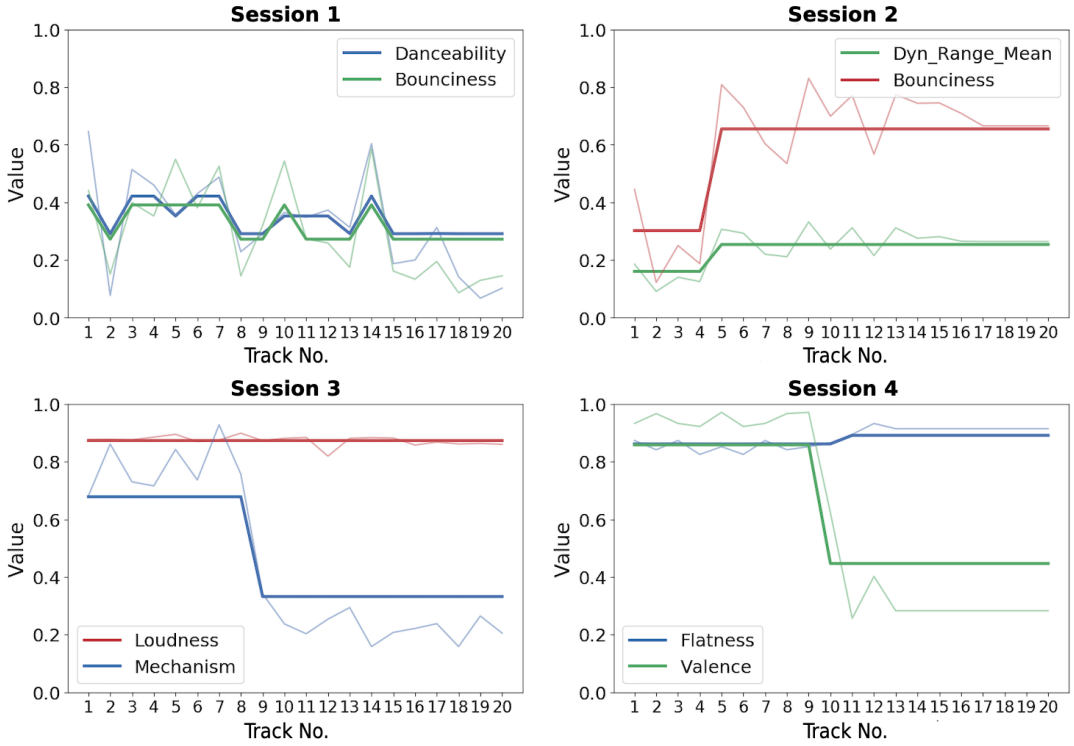



FIGURE 3.6: Raw audio feature values (Normal line) and their corresponding states (Bold line) for 2 selected audio features in 4 sample listening sessions

### 3.4.2 Different audio attributes have different number of states and state transitions

Even though we specify up to 10 latent states in our state extraction model, we have found that none of the audio attributes in any session end up with more than 4 distinct states. In fact, most attributes either have a single state (i.e. no notable fluctuation in a session) or only 2 states, with a minority of cases exhibiting 3 or 4 states.

Figure 3.7 shows that many features have at least 20% of the sessions with 2 or more states. However, a few features (e.g. time_signature, instrumentalness, key) have fewer than 10% of the sessions with 2 or more states. Upon closer analysis, these features have bimodal or highly skewed distributions, which could be a reason for the lack of fluctuations in their values.

When measuring transitions between states, Figure 3.8 demonstrates that

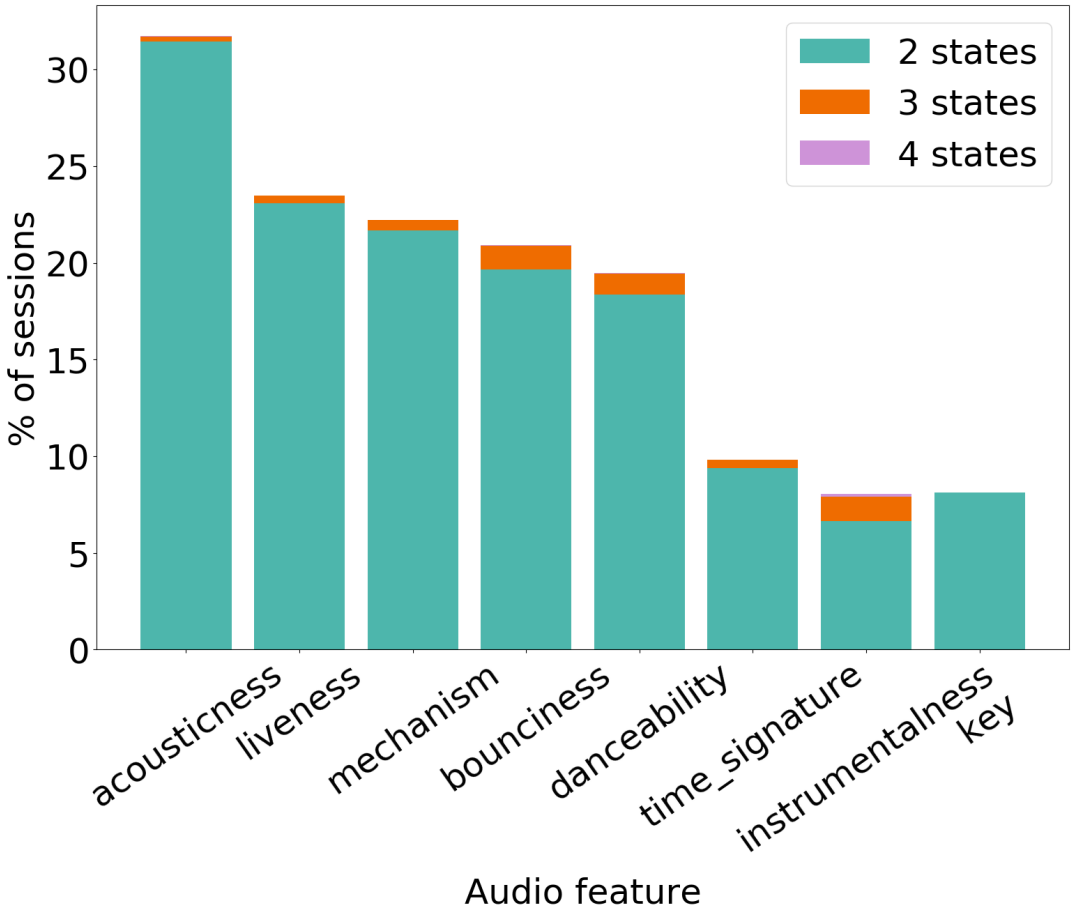



FIGURE 3.7: Percentage of sessions with 2,3,4 states respectively across selected audio features

different audio attributes have different levels of state transitions. Here, a state transition refers to a track transition that corresponds to a change in state. By comparing the ratio of state transitions to track transitions across all sessions, we notice that attributes such as 'acousticness' and 'liveness' exhibit a lot more state transitions than others like 'time_signature' or 'instrumentalness'. Hence, there are differences in audio attributes that vary across different features, in terms of number of states and transitions.

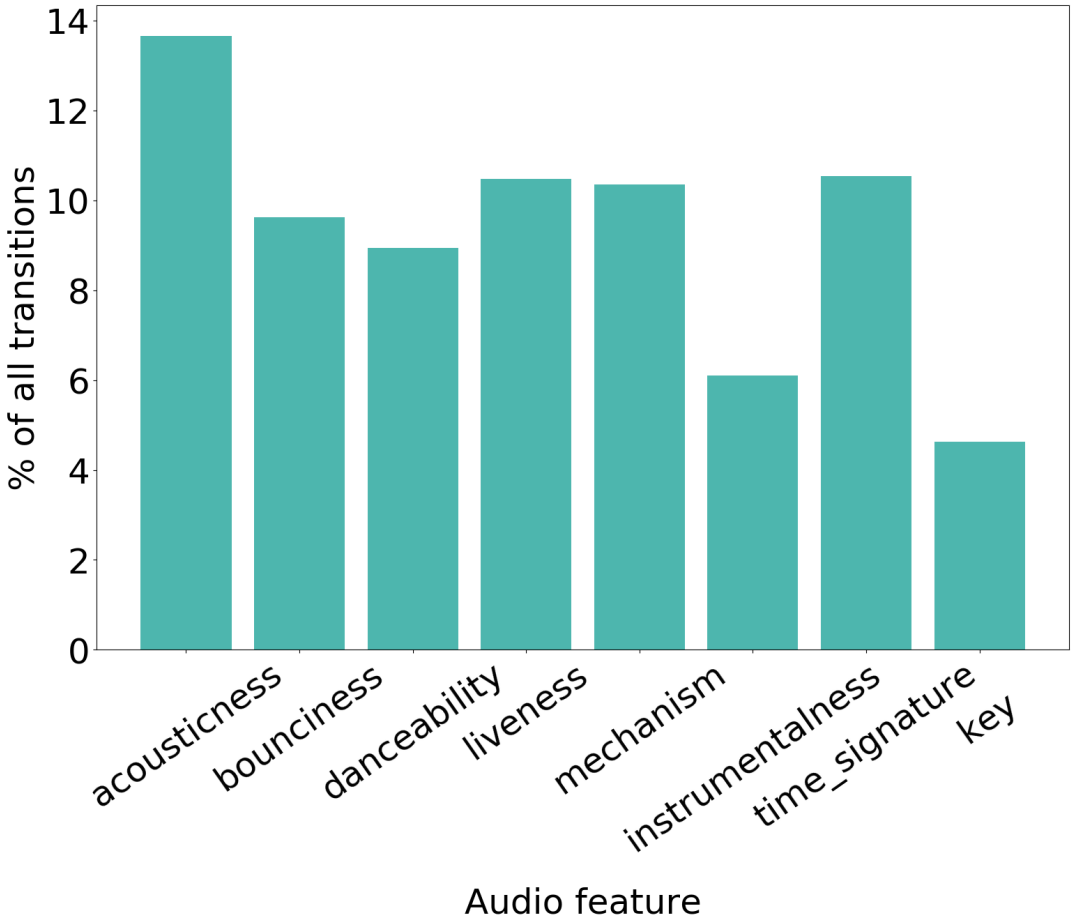



FIGURE 3.8: Percentage of all transitions that are state transitions across selected audio features

### 3.4.3 States and state transitions in audio attributes are correlated with user satisfaction

Measuring state transitions are important since there is a possibility that the user may be satisfied or dissatisfied with the new audio state that they are presented with. While we observed that some features have more state transitions than others in the previous section, we found that there is a steady trend when we compare the ratio of skip/non-skip transitions to all state transitions. Here, a skip/non-skip transition refers to a state transition between two tracks that not only corresponds to a change in state, but also a change in the skip behaviour of the track. The ratio is $\sim$25% across all audio attributes, which is in fact a significant percentage of all state transitions.

The number of state transitions in a session can potentially impact user satisfaction, as users may find it awkward when audio properties of tracks changes frequently in their sessions. Figure 3.9 compares the average number of state transitions in the top 5 features of SAT and DSAT sessions. Here, the top 5 features are chosen based on the number of state transitions the feature has in total across all sessions. We found that SAT sessions do have fewer state transitions in general, which is in line with our expectations.

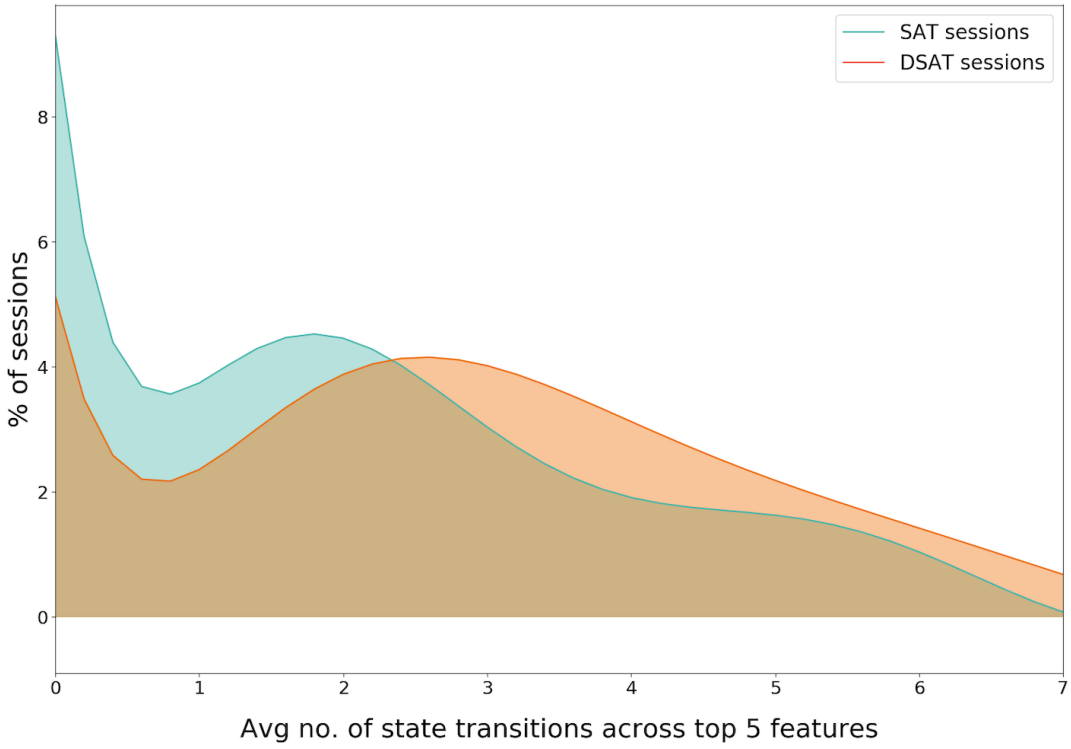



FIGURE 3.9: Distribution of sessions across average number of state transitions in top 5 features

Lastly, we attempt to compare the states of audio features we extracted with that of user satisfaction. We observe that 40 to 50% of the extracted states in audio features have useful information, as presented in Table 3.2, as they exhibit correlation with SAT/DSAT. This finding means that there is a potential use case for sessions to be optimised for SAT if we re-rank tracks with SAT states over tracks with DSAT states. Nevertheless, this may be pure coincidence, as there might be other reasons that can lead to certain states being correlated with SAT/DSAT.

TABLE 3.2: Percentage of states over all sessions that are SAT or DSAT for selected audio features

| | acousticness | bounciness | danceability | key | tempo |
|---|---|---|---|---|---|
| **SAT** | 23.70 | 21.73 | 21.90 | 23.23 | 23.97 |
| **DSAT** | 25.36 | 26.50 | 26.67 | 25.96 | 24.06 |

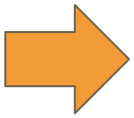
### 3.4.4 Insights about audio states can help to improve track sequencing for user satisfaction

#### Counterfactual re-ranking experiment

We have since shown that correlation exists between audio states that we extracted and user satisfaction. We hypothesise that the extracted states can be leveraged to optimise listening sessions for SAT. To investigate this research question, a counterfactual track re-ranking experiment is designed and performed to confirm our hypothesis. By considering the skip behaviour of all the tracks and the states of all the audio attributes in a session, we pick the best audio attribute that can produce a track sequence that ranks not-skipped tracks higher than skipped tracks. Performing this experiment would therefore allow us to investigate the scope of potential improvement possible, if and only if we know the audio feature that best characterises each session.

Specifically, for each session, we pick the top audio attribute that help us re-rank the tracks best. The top audio attribute would have 2 or more states (we wouldn't be able to re-rank the tracks if an audio attribute has no variations), wherein we have to define which state is better than other states of the same attribute. To quantify the 'good'ness of a state, we use the average number of skips of tracks belonging to the state – a 'good' state would have lower average number of skips than a 'bad' state. Next, we re-rank the tracks in a session by recommending tracks in 'good' states before tracks in 'bad' states. The original session position of the track is used as a tie-breaker for tracks in the same state, since tracks in a session in MSSD have an implicit ranking based on similarity between the user and the track.

| Acousticness | | |
|---|---|---|
| **Track** | **State** | **Non-skip** |
| A | 1 | 1 |
| B | 2 | 0 |
| C | 1 | 1 |
| D | 2 | 1 |
| E | 2 | 0 |
| F | 2 | 0 |
| G | 1 | 1 |



| Acousticness | | |
|---|---|---|
| **Track** | **State** | **Non-skip** |
| A | 1 | 1 |
| C | 1 | 1 |
| G | 1 | 1 |
| B | 2 | 0 |
| D | 2 | 1 |
| E | 2 | 0 |
| F | 2 | 0 |

FIGURE 3.10: Illustration of re-ranking mechanism for the 'acousticness' attribute in a sample session in the counterfactual experiment. On the left, there is a sample session with 7 tracks, with information of the extracted states of the 'acousticness' attribute and the non-skip behaviour. It can inferred that state 1 is better than state 2, since all 3 tracks in state 1 are not skipped, but only 1 out of 4 tracks in state 2 are not skipped. Thus, we can rank tracks in state 1 higher than all the tracks in state 2, where tracks in the same state are ranked by its original session position, giving us the re-ranked session on the right.

Figure 3.10 illustrates the re-ranking mechanism for a single attribute in a session. However, we perform this re-ranking step for all audio attributes in a session and select the top attribute that gives us the best ranking score, using NDCG@10.

### Results and Analysis

First, we start by analysing the top few audio attributes across all sessions, with the distributions presented in Figure 3.11. We can immediately note that there are different audio attributes that best optimise different listening sessions. Even though 'acousticness' and 'beat_strength' appear more often than other features, no single feature dominates all the sessions. The main

takeaway here is: *there is no single audio attribute which will work best for all sessions*. This strongly motivates the need to develop models that are able to predict the top audio attribute for each session, which can subsequently be used to optimise sessions for user satisfaction.

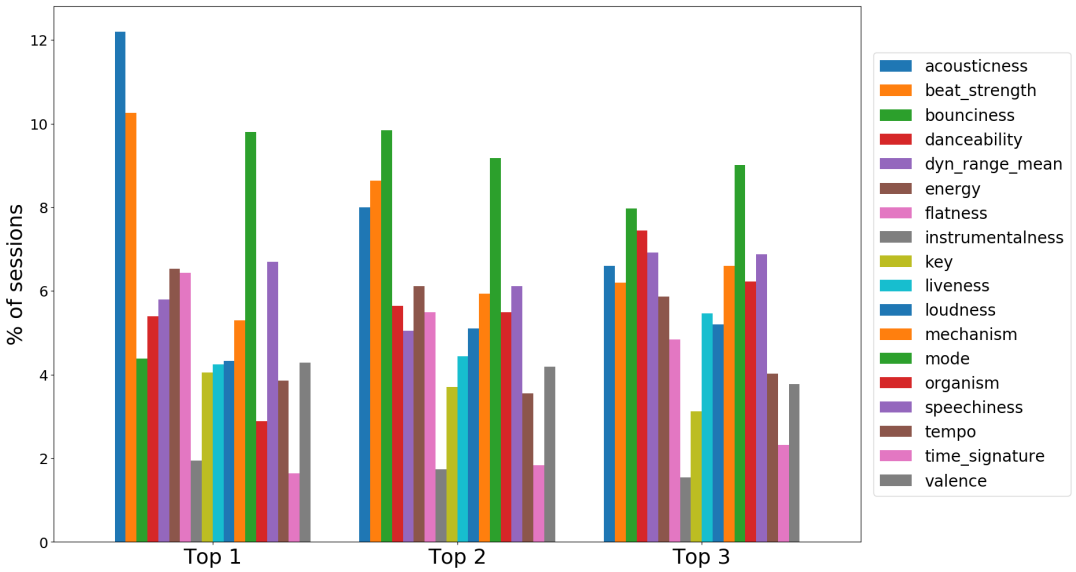



FIGURE 3.11: Distribution of audio attributes amongst the top 1/2/3 attributes of sessions

Next, we have found that if we know the top audio attribute and the associated state best suited for each session, we are indeed able to optimise sessions for user satisfaction. Table 3.3 lists the results of all our experiments, evaluated using NDCG@10, AP@10, P@10 and RR@10. If we use a random feature, or the most common top audio attribute identified in Figure 3.11 (i.e. 'acousticness'), we do not improve the ranking performance by much. However, if we had access to an oracle that provided us with the best audio attribute and the associated state that is most suited to each session, we are able to improve the ranking performance by over 20% relative to ranking by user relevance.

This result indicates that the potential in leveraging audio states for improved track sequencing is significant. By developing a prediction model that can accurately predict the top audio attribute and corresponding states

TABLE 3.3: Results of counterfactual track re-ranking experiment on 10,000 sessions. **Rank by User Relevance**: Rank by user-track similarity (i.e. sessions in its current state). **Random Top Attribute**: Rank using a randomly selected audio attribute. **Global Top Attribute**: Rank using the audio attribute that appear most frequently as the top attribute across all sessions (i.e. 'acousticness'). **Known Top Attribute**: Rank using the best audio attribute for each session.

| | NDCG@10 | AP@10 | P@10 | RR@10 |
|---|---|---|---|---|
| Rank by User Relevance | 0.654 | 0.692 | 0.506 | 0.790 |
| Random Top Attribute | 0.672 | 0.702 | 0.520 | 0.800 |
| Global Top Attribute | 0.677 | 0.705 | 0.523 | 0.803 |
| Known Top Attribute | **0.796** | **0.835** | **0.604** | **0.938** |

which best characterise each session, we can improve user SAT metrics by a significant margin, leading to enhanced user satisfaction in streaming sessions, which is the focus of the second part of our research.

# 4 Leveraging Audio States & Transitions for Track Sequencing

In this chapter, we focus on the second part of our research. We start by motivating the need to develop a track sequencing model that leverages audio states & transitions. We define some of the issues that can arise when developing such a model, before proposing our track re-ranking model. Lastly, we present results of our experiments and evaluate the model's performance across different subsets of streaming sessions.

## 4.1 Motivation

Having established that audio attributes vary within music streaming sessions, and that audio states and transitions are indeed correlated with user satisfaction, we have set out to show that those insights can help to improve track sequencing for user satisfaction. We conducted a counterfactual track re-ranking experiment, assuming full knowledge of the top audio attribute and the preferred state for each session. From the experiment, we demonstrated that if we had access to an oracle that gave us the audio attribute and corresponding state that best characterises each session, we can optimise user satisfaction metrics by over 20%. When we analyse the distribution of top audio attributes amongst all the sessions, we also learnt that there is no single audio attribute that will work best for all sessions. Our findings drive the

need to come up with an accurate audio attribute prediction model, which can be leveraged for improved track sequencing in sessions.

## 4.2 Problem Statement

In our counterfactual track re-ranking experiment, we determined the top audio attribute per session and used that to re-rank tracks by prioritising the best state of that attribute. Determining the best state of the attribute in turn uses the average number of skips in the states. This means that we would require access to the user's skip behaviour when sequencing tracks. However, that would not be possible when we want to re-rank a collection of unseen tracks, since we do not have access to the skip behaviour of those tracks. Moreover, we will be using the skip behaviour of a session to evaluate the ranking performance of our track sequencing model.

Hence, in the pursuit of a track re-ranking model that leverages audio states and transitions, we need to develop a way to predict the top audio attribute per session and determine which of the states in that attribute is preferred, *without knowledge of the skip behaviour*. We have also assumed thus far that each session is best served by one audio attribute, but we can potentially consider 2 or 3 audio attributes, which may serve as a better characterisation of sessions. We can also incorporate user interaction feedback from the first few tracks of each session to make an intelligent guess about the preferred audio attributes and states for that session, and use that to re-rank the remaining tracks in the session.

## 4.3 Proposed Track Re-Ranking Model

Considering the problems discussed and some potential ideas, we propose the following track re-ranking model, which are divided into three steps:

1. **Audio Attribute Prediction** – *Predict which audio attributes to use to re-rank a session* (Section 4.3.1)

2. **Preferred State Prediction** – *Determine the preferred states in the selected audio attributes* (Section 4.3.2)

3. **Track Re-ranking Algorithm** – *Re-rank the candidate tracks using the selected audio attributes and their preferred states* (Section 4.3.3)

### 4.3.1 Audio Attribute Prediction

In the first step, we develop a classification model that can predict the top 3 audio attributes of each session. By manually engineering features that characterise different aspects of each streaming session, we can represent each session using a session vector to be passed on to a classification model that can predict the top few audio attributes for that session. In that regard, we have come up with a list of features that we can compute for each session:

(a) Session length

(b) Number of audio attributes with 2 or more states

(c) Number of tracks in each state (for each audio attribute)

(d) Number of state transitions (for each audio attribute)

(e) Number of state transitions in the first N in-session tracks (for each audio attribute)

(f) Number of state transitions that coincide with skip/non-skip transitions in the first N in-session tracks (for each audio attribute)

(g) Number of non-skip tracks in the first N in-session tracks (for each audio attribute)

(h) Value count of best audio attribute labels for the 10 nearest sessions

Feature (f) taps on the idea of incorporating user interaction feedback, by considering the skip behaviour of the first N in-session tracks and its correlation with state transitions. In feature (h), we find the nearest sessions for each session by computing the cosine similarity between each session and all other sessions, where each session is represented as a concatenation of features (a)-(g). The top audio attribute labels of the 10 nearest sessions are chosen and we take a count of unique occurrences of the labels to form feature (h). Feature (h) is then appended to the concatenation of features (a)-(g) to form the final session vector.

We have chosen to use a Gradient Boosting Machine (GBM) [16] as our classifier, as it has been demonstrated empirically on various competitive data science platforms to be a robust out-of-the-box classifier that can perform well on a wide variety of structured data. Specifically, we use the LightGBM [26] implementation in our experiments.

To train the classifier, we prepared a training set of 250,000 streaming sessions from MSSD, which is further split into a train and validation set through a 80%/20% split. The test set contains 50,000 sessions taken from a time period different from sessions in the training set, to ensure our model generalises well to sessions across various time periods. Similar to the experimental setup in the first part of our research, sessions that are in shuffle mode, or containing different listening contexts, are not included in our dataset. Grid search is performed on various hyperparameters, such as the number of boosted trees (n_estimators), using the validation set to obtain the optimal hyperparameter settings. N was set to 5 (i.e. we consider only the first 5 in-session tracks) for features (e), (f) and (g).

### 4.3.2 Preferred State Prediction

Once the top few audio attributes per session are predicted, we would also need to determine which states in those attributes are likely to be preferred by the user. To do so, we design a heuristic to incorporate skip feedback from the first N in-session tracks. Specifically, we compute a score for each state in each audio attribute, by assigning a positive score (+1) if the state appeared in a non-skipped track, and a negative score (-1) if the state appeared in a skipped track. The state with the highest score in each audio attribute after considering the first N in-session tracks would be the preferred state for that attribute. The aforementioned scoring mechanism is illustrated in Figure 4.1.

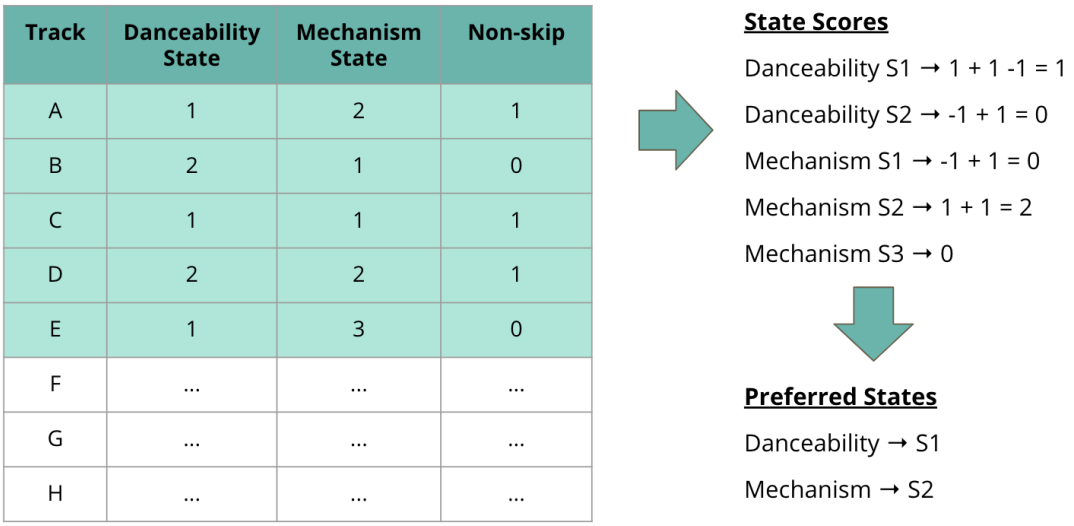



FIGURE 4.1: Illustration of preferred state prediction using first N=5 in-session tracks of a sample session, focusing on the top 2 audio attributes ('danceability' and 'mechanism'). e.g. State 1 (S1) of 'danceability' appeared in 3 tracks, of which 2 were not skipped, producing a final score of 1. It is the preferred state for 'danceability' since its score is higher than that of S2.

### 4.3.3 Track Re-ranking Algorithm

With the top audio attributes and their preferred states for each session, we can consider various techniques to perform track re-ranking in streaming sessions. Note that we should only be *re-ranking the remaining tracks in each session*, since the skip behaviour of the first N in-session tracks have been taken into consideration in the first 2 steps of the ranking model. As such, we propose the following two techniques:

1. **M1: Rank by preferred state with tie-breaking criteria**. Similar to the technique that was used in the counterfactual track re-ranking experiment (Chapter 3.4.4), we rank tracks in the preferred state higher than tracks in other states, while tracks in the same state are tie-breaked by a second criteria. The second criteria can be user relevance (session position), audio similarity (cosine similarity with the mean of first N in-session tracks) or popularity ('us_popularity_estimate' in MSSD).

2. **M2: Average score with audio similarity and/or popularity**. Using a tie-breaking mechanism leads to a situation where tracks not in the preferred states will always be 'stuck' below those in the preferred states, since we use audio states as the primary feature. Instead of tie-breaking, we score each remaining track using an average of its state score (computed in Step 2 of the ranking model) and its audio similarity score or popularity score or both, which we use to re-rank all the remaining tracks instead.

For each of these proposed technique, besides considering just the top 1 audio attribute, we also consider the top 2 or 3 predicted audio attributes per session. Specifically, for the top 2 or 3 audio attributes, we sum the state scores across the attributes before re-ranking.

Thus, our proposed ranking model tackles the problem of determining the top audio attributes and their preferred states for each session through

a classifier that uses a variety of manually engineered features representing a streaming session and the incorporation of skip feedback. The model considers the top few audio attributes per session instead of only one, and also utilises other track attributes such as similarity and popularity.

## 4.4 Experimental Results

In this section, we begin by evaluating the performance of the audio attribute prediction model which we use in the first step of the ranking model, followed by the performance of the complete track ranking model.

### 4.4.1 Audio Attribute Prediction

We present the results of the audio attribute prediction classification model in Table 4.1. Grid search was performed using the validation set to determine the optimal number of boosted trees to use in our LightGBM model, where 100 trees performed the best out of a range of 25-500 trees. N was set to 5 during feature engineering (i.e. we only use the first 5 in-session tracks).

TABLE 4.1: Results of top audio attribute prediction model using a LightGBM classifier with 100 boosted trees. Note: a 'soft' accuracy and macro-averaged F1 is being used for Top 2 and 3, i.e. the model has predicted correctly if the true audio attribute appears in the predicted top 2 or 3 audio attributes.

| | **Accuracy** | **Macro F1** |
|---|---|---|
| Top 1 | 0.426 | 0.422 |
| Top 2 | 0.651 | 0.649 |
| Top 3 | 0.784 | 0.783 |

Our classifier achieves 42% accuracy in predicting the top audio attribute for each session. If we relax the accuracy criteria to the top 3 audio attributes being predicted by the classifier, we can achieve up to 78% accuracy, which highlights the room for improvement for the classifier. The same trend can be observed using macro-averaged F1 score.

### 4.4.2 Overall Track Re-ranking

The results of the complete track re-ranking model using both proposed re-ranking techniques are presented in Table 4.2. First, we observe that both M1 and M2 and their variants already improve upon the user relevance baseline. Using audio similarity in both M1 and M2 helps to improve the ranking performance further, surpassing that of the audio similarity baseline.

Likewise, using popularity in both M1 and M2 also improves the ranking performance, but does not surpass the strong popularity baseline. However, it is noteworthy that even though recommending popular content seems to improve ranking performance, it suffers from the filter bubble effect of popularity bias [44]. In that regard, if we compare Popularity@K between our proposed models and the popularity baseline, our models score lower. This means that our model are less likely to recommend popular content, while still maintaining competitive performance with the popularity baseline on user satisfaction metrics, which is an important result.

Moreover, if we knew in advance the correct top audio attributes for each session (Known Attr), but utilise our proposed preferred state prediction and track ranking techniques, we are able to outperform all our baselines. Not only does this motivate the need to improve our attribute prediction model, this also validates the efficacy of Steps 2 and 3 of the track re-ranking model. Knowing both the top audio audio attribute and their preferred states (Known Attr + States) can improve user satisfaction metrics even further, which suggests there is still room for improvement in our preferred state prediction heuristics.

Lastly, considering the top 2 and 3 audio attributes per session across our models offers slight improvement or equivalent performance in user satisfaction metrics, with the exception of M1 (Pop). This indicates the characterisation of sessions by more than 1 audio attribute has its benefits. However,

TABLE 4.2: Ranking results on remaining candidate tracks in 50,000 sessions, after taking out the first $N = 5$ in-session tracks from each session. We define 3 baseline models, where we rank solely by user relevance, audio similarity or popularity (n.b. we will refer to these as User, Audio or Pop subsequently). Brackets indicate the tie-breaking criteria for M1, or the score with which the state scores are averaged with for M2. *Known Attr (+ States)* refer to the hypothetical scenarios where we have 100% accuracy in predicting the top 3 audio attributes (and preferred states), but follow M1 (User) in re-ranking the remaining candidate tracks. Popularity@K measures the average popularity score of the first K tracks.

| | **NDCG@10** | | | **AP@10** | | |
|---|---|---|---|---|---|---|
| | **Top 1** | **Top 2** | **Top 3** | **Top 1** | **Top 2** | **Top 3** |
| User Relevance | 0.667 | – | – | 0.614 | – | – |
| Audio Similarity | 0.700 | – | – | 0.642 | – | – |
| Popularity | 0.719 | – | – | 0.661 | – | – |
| M1 (User) | 0.687 | 0.689 | 0.693 | 0.630 | 0.632 | 0.635 |
| M1 (Audio) | 0.708 | 0.707 | 0.707 | 0.650 | 0.649 | 0.650 |
| M1 (Pop) | 0.716 | 0.714 | 0.712 | 0.658 | 0.655 | 0.654 |
| M2 (Audio) | 0.707 | 0.707 | 0.707 | 0.648 | 0.648 | 0.649 |
| M2 (Pop) | 0.716 | 0.718 | 0.717 | 0.658 | 0.659 | 0.659 |
| M2 (Audio + Pop) | 0.717 | 0.718 | 0.718 | 0.658 | 0.661 | 0.661 |
| Known Attr | 0.729 | 0.734 | 0.734 | 0.681 | 0.682 | 0.681 |
| Known Attr + States | 0.760 | 0.778 | 0.788 | 0.713 | 0.731 | 0.742 |
| | **Popularity@5** | | | **Popularity@3** | | |
| | **Top 1** | **Top 2** | **Top 3** | **Top 1** | **Top 2** | **Top 3** |
| User Relevance | 0.264 | – | – | 0.264 | – | – |
| Audio Similarity | 0.267 | – | – | 0.268 | – | – |
| Popularity | 0.325 | – | – | 0.353 | – | – |
| M1 (User) | 0.264 | 0.264 | 0.264 | 0.264 | 0.264 | 0.264 |
| M1 (Audio) | 0.265 | 0.265 | 0.265 | 0.267 | 0.266 | 0.265 |
| M1 (Pop) | 0.292 | 0.286 | 0.283 | 0.314 | 0.298 | 0.293 |
| M2 (Audio) | 0.265 | 0.265 | 0.266 | 0.267 | 0.267 | 0.267 |
| M2 (Pop) | 0.293 | 0.301 | 0.303 | 0.315 | 0.320 | 0.322 |
| M2 (Audio + Pop) | 0.294 | 0.298 | 0.298 | 0.311 | 0.315 | 0.316 |
| Known Attr | 0.265 | 0.265 | 0.265 | 0.265 | 0.265 | 0.265 |
| Known Attr + States | 0.265 | 0.266 | 0.266 | 0.265 | 0.266 | 0.266 |

the improvement offered by multiple audio attributes is even more distinct if we look at the jump in user satisfaction metrics of the ideal baseline (Known Attr + States), which implies more can be done to exploit multiple audio attributes in characterising sessions.

## 4.5 Analysis

Having evaluated the experimental results of our proposed track re-ranking model in comparison to various baselines, we analyse the performance of the model across different subsets of streaming sessions, through criteria such as session length, average number of skips, context type and reason for track start. For each criteria, we compare the NDCG@10 scores of the user relevance baseline, our model, the difference between our model and the user relevance baseline, and the known top attribute baseline.

#### Session Length

Our model performs better on shorter session lengths, as shown in Table 4.3, where the increase in ranking performance for sessions with 10-17 tracks is larger than sessions with 18-20 tracks. One possible explanation is that longer streaming sessions are harder to optimise, since user preferences are more likely to change over the period of the session when the session length is longer.

TABLE 4.3: Ranking results (NDCG@10) on divided across different session lengths.

| Session Length | % sessions | User Rel | Model | Δ | Known |
|---|---|---|---|---|---|
| 10-13 | 32.438 | 0.781 | 0.806 | +0.025 | 0.828 |
| 14-17 | 18.726 | 0.685 | 0.713 | +0.028 | 0.745 |
| 18-20 | 48.836 | 0.585 | 0.598 | +0.014 | 0.658 |

#### Average number of skips

In Table 4.4, we split the sessions by the average number of skips observed across the tracks. The increase in performance that our model offers is higher when the average number of skips is between 0.25 and 0.75. This is in line with expectations since it is generally harder to optimise sessions which tracks

are mostly skipped, as it could be that all the candidate tracks being recommended are just not well received by the user. Similarly, sessions with tracks that are mostly not skipped are likely to indicate lean-back listening modes, such as sleep playlists, where user interactions are unlikely.

TABLE 4.4: Ranking results (NDCG@10) on divided across different average number of skips.

| Avg skips | % sessions | User Rel | Model | Δ | Known |
|---|---|---|---|---|---|
| 0.0 - 0.25 | 29.576 | 0.929 | 0.940 | +0.011 | 0.947 |
| 0.25 - 0.5 | 23.205 | 0.704 | 0.729 | +0.025 | 0.757 |
| 0.5 - 0.75 | 25.975 | 0.553 | 0.579 | +0.027 | 0.633 |
| 0.75 - 1.0 | 21.244 | 0.402 | 0.422 | +0.020 | 0.514 |

### Context Type

MSSD offers information on the listening context of the streaming sessions, which we analyse in Table 4.5. Our model does not perform particularly well in radio sessions, personalised playlists and charts. This could be due to the fact that tracks in these session contexts are already acoustically similar. For example, radio sessions are generated based on particular artists or albums, which are likely to create sessions with tracks that are acoustically similar.

TABLE 4.5: Ranking results (NDCG@10) on sessions divided across different context types.

| Context Type | % sessions | User Rel | Model | Δ | Known |
|---|---|---|---|---|---|
| user_collection | 44.151 | 0.656 | 0.676 | +0.021 | 0.722 |
| radio | 19.931 | 0.652 | 0.664 | +0.012 | 0.709 |
| catalog | 16.505 | 0.695 | 0.724 | +0.029 | 0.758 |
| editorial_playlist | 13.819 | 0.667 | 0.689 | +0.022 | 0.726 |
| personalized_playlist | 4.205 | 0.752 | 0.770 | +0.018 | 0.799 |
| charts | 1.389 | 0.670 | 0.666 | -0.004 | 0.728 |

**Reason for track start**

MSSD also offers information on track context, specifically the action that leads to the start of each track. We summarise each session by the most common reason and analyse them in Table 4.6. Our model does not perform as well on sessions with mostly track dones, as these sessions are also likely to be indicators of lean-back listening sessions. Sessions with mostly click rows are also hard to optimise. If users are selecting the songs manually in these sessions, it could be due to the intent to look for specific tracks rather than the intent to find acoustically similar tracks.

TABLE 4.6: Ranking results (NDCG@10) on sessions divided across different reasons for track start.

| Reason | % sessions | User Rel | Model | Δ | Known |
|---|---|---|---|---|---|
| fwdbtn | 52.591 | 0.554 | 0.576 | +0.022 | 0.637 |
| trackdone | 34.933 | 0.874 | 0.886 | +0.012 | 0.899 |
| backbtn | 9.119 | 0.520 | 0.571 | +0.051 | 0.612 |
| clickrow | 3.335 | 0.679 | 0.682 | +0.003 | 0.735 |

# 5 Conclusion

In this thesis, we have leveraged audio characteristics of tracks within a streaming to improve track sequencing in online music streaming sessions. We have done so in two parts: first, we investigated how audio states and transitions can be extracted from audio attributes in streaming sessions and their impact on user satisfaction, and second, we show that these audio states and transitions can be leveraged for track sequencing to improve user satisfaction metrics. We begin this concluding chapter by summarising the main findings of both parts of our research, and highlighting the contributions we made. The implications of our findings are discussed, and we reflect on few promising future research directions.

## 5.1 Main Findings

The first part revolved around understanding audio attributes in streaming sessions. We attempt to quantify how audio characteristics vary in streaming sessions, by proposing a state extraction model using HMMs to find underlying representations of the changes in audio attributes in streaming sessions. We set out 4 research questions to investigate, and through our analysis, we found that:

1. Audio characteristics do vary across tracks in a session, and fluctuations in audio characteristics are fairly common.

2. Different audio attributes have different number of states and state transitions.

3. States and state transitions in audio attributes are indeed correlated with user satisfaction.

4. Through a counterfactual track re-ranking experiment, we established that insights about audio states can indeed help in track sequencing for improved user satisfaction.

Most importantly, we show that leveraging the best audio attribute and the preferred state per session to re-rank tracks can improve user satisfaction metrics by over 20%.

In the second part, we motivated the need to develop a prediction model that can accurately predict the top audio attribute and corresponding states which best characterise each session. We proposed a three-step track re-ranking model that incorporates the aforementioned prediction model, a mechanism to determine the preferred states, and a track sequencing algorithm that utilises the predicted audio attributes and preferred states. Our experimental results show that our proposed model can indeed optimise user satisfaction metrics in streaming sessions, while not over-exposing popular content. Analysing the performance of our model across different types of sessions indicate that our model performs well on shorter sessions with skip behaviour that are not too extreme. Our model also performs well on common session contexts, with the exception of sessions correlated with lean-back listening modes, or sessions that only contain acoustically similar tracks.

## 5.2 Contributions

Through our research in the thesis, we have made two important contributions to the body of work on music recommender systems:

1. We proposed a state extraction model, through a multiple changepoint detection technique that utilises HMMs. The model not only considers

local variations of audio attributes in streaming sessions, but also incorporate global variations through the hyperparameters. Applying this model on a dataset of music streaming sessions have shown that audio states and transitions are indeed correlated with user satisfaction. We highlight that leveraging such information about state transitions holds promise, as it can help us improve key user satisfaction metrics, which have implications on the design of sequential recommendation techniques.

We have published a conference paper as a result of this body of work: *Aaron Ng and Rishabh Mehrotra. Investigating the Impact of Audio States & Transitions for Track Sequencing in Music Streaming Sessions. In Fourteenth ACM Conference on Recommender Systems (RecSys '20).*

2. We also proposed a track re-ranking model that leverages the extracted audio states and transitions to better sequence tracks in streaming sessions. The three-part model uses an audio attribute prediction model, heuristics to determine the preferred states, before combining these predictions to re-rank tracks. Applying this model a dataset of music streaming sessions have shown that we are able to improve user satisfaction in sessions. Our proposed model performs competitively with respect to other baselines without over-exposing popular content.

## 5.3 Implications

Our findings highlight the importance of leveraging audio characteristics for building track sequencing models and recommendation systems for audio streaming sessions. Firstly, we encourage the development of a real-time track sequencing model that incorporates feedback from recent user

interactions. Such a model can enable the development of highly personalised recommendations on music streaming platforms. Next, we advocate for track sequencing models that help maintain the "flow" of audio aesthetics in streaming sessions. If tracks are sequenced in a manner with smooth transitions in audio properties from one to another, we can produce listening sessions with less abrupt transitions, leading to improve user satisfaction. Lastly, we hope to inspire the development of intent extraction methods that can help identify audio attributes that best characterise each streaming session, thereby supporting the current intent of the user.

## 5.4 Future Work

We have presented various important implications of our research in this thesis. However, there are a number of opportunities for future work. First, we can consider more advanced architectures for the development of the audio attribute prediction classifier, in the initial step of our track re-ranking model. The idea of designing deep learning models defined on sets rather than the traditional approach of using fixed dimensional vectors, as proposed by [55], has spurred the development of sets-based architecture in recent years. If we consider a streaming session as a set of music tracks, applying models that consider a group of tracks rather than a condensed representation of a session can perhaps lead to improved performance in the audio attribute prediction task. Such an architecture can also potentially eliminate the step of manually engineering features to represent a session, if it can learn the sequential interactions between tracks in sessions.

Next, we should consider obtaining and incorporating more data, specifically user-related features, that are not currently available in the dataset being used in our research. Every user has different music preferences and listening habits, which in turn affects their interaction behaviour on music

streaming platforms. Understanding and utilising user preferences can play an important role in furthering the development of track sequencing models.

Finally, we can be more selective about the type of audio attributes to use. In our research, we asserted that each audio attribute is independent and equally important in music recommendation. However, survey studies of users of music streaming platforms have shown that this may not be true: the presence of vocals and the ability for music to evoke emotions have been found to be more important than other audio attributes such as tempo or harmony [12]. The understanding of the interplay between audio characteristics and people's musical taste is therefore a promising area for further research.

# Bibliography

[1] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. "Mining Association Rules between Sets of Items in Large Databases". In: *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*. SIGMOD '93. Washington, D.C., USA: Association for Computing Machinery, 1993, 207–216. ISBN: 0897915925. DOI: 10.1145/170035.170072.

[2] N. Ahmed, T. Natarajan, and K. R. Rao. "Discrete Cosine Transform". In: *IEEE Transactions on Computers* C-23.1 (1974), pp. 90–93.

[3] Luke Barrington et al. "Combining Feature Kernels for Semantic Music Retrieval". In: *ISMIR 2008, 9th International Conference on Music Information Retrieval, Drexel University, Philadelphia, PA, USA, September 14-18, 2008*. Ed. by Juan Pablo Bello, Elaine Chew, and Douglas Turnbull. 2008, pp. 614–619. URL: http://ismir2008.ismir.net/papers/ISMIR2008_160.pdf.

[4] James Bennett, Stan Lanning, et al. "The Netflix Prize". In: *Proceedings of KDD Cup and Workshop*. Vol. 2007. 2007, p. 35.

[5] Geoffray Bonnin and Dietmar Jannach. "Automated Generation of Music Playlists: Survey and Experiments". In: *ACM Comput. Surv.* 47.2 (Nov. 2014). ISSN: 0360-0300. DOI: 10.1145/2652481.

[6] Brian Brost, Rishabh Mehrotra, and Tristan Jehan. "The Music Streaming Sessions Dataset". In: *The World Wide Web Conference*. WWW '19.

San Francisco, CA, USA: Association for Computing Machinery, 2019, 2594–2600. ISBN: 9781450366748. DOI: 10.1145/3308558.3313641.

[7] Michael A. Casey et al. "Content-Based Music Information Retrieval: Current Directions and Future Challenges". In: *Proceedings of the IEEE* 96.4 (2008), pp. 668–696.

[8] Zeina Chedrawy and Syed Sibte Raza Abidi. "A Web Recommender System for Recommending, Predicting and Personalizing Music Playlists". In: *Web Information Systems Engineering - WISE 2009*. Ed. by Gottfried Vossen, Darrell D. E. Long, and Jeffrey Xu Yu. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 335–342. ISBN: 978-3-642-04409-0.

[9] Ching-Wei Chen, Kyogu Lee, and Ho-Hsiang Wu. "Towards a Class-Based Representation of Perceptual Tempo for Music Retrieval". In: *2009 International Conference on Machine Learning and Applications*. 2009, pp. 602–607.

[10] Zhi-Sheng Chen and Jyh-Shing R. Jang. "On the Use of Anti-Word Models for Audio Music Annotation and Retrieval". In: *IEEE Transactions on Audio, Speech, and Language Processing* 17.8 (2009), pp. 1547–1556.

[11] Sally Jo Cunningham, David Bainbridge, and Annette Falconer. "'More of an Art than a Science': Supporting the Creation of Playlists and Mixes". In: *ISMIR 2006, 7th International Conference on Music Information Retrieval, Victoria, Canada, 8-12 October 2006, Proceedings*. 2006, pp. 240–245.

[12] Andrew Demetriou et al. "Vocals in Music Matter: the Relevance of Vocals in the Minds of Listeners". In: *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*. Ed. by Emilia G 'o mez et al. 2018, pp. 514–520. URL: http://ismir2018.ircam.fr/doc/pdfs/98_Paper.pdf.

[13] J. Stephen Downie. "Music Information Retrieval". In: *Annual review of information science and technology* 37.1 (2003), pp. 295–340.

[14] J. Stephen Downie et al. "The Music Information Retrieval Evaluation eXchange: Some Observations and Insights". In: *Advances in Music Information Retrieval*. Ed. by Zbigniew W. Raś and Alicja A. Wieczorkowska. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 93–115. ISBN: 978-3-642-11674-2. DOI: 10.1007/978-3-642-11674-2_5.

[15] Adrian Freed. "Music metadata quality: a multiyear case study using the music of skip james". In: *Audio Engineering Society Convention 121*. Audio Engineering Society. 2006.

[16] Jerome H Friedman. "Greedy function approximation: a gradient boosting machine". In: *Annals of statistics* (2001), pp. 1189–1232.

[17] Zhouyu Fu et al. "On Feature Combination for Music Classification". In: *Structural, Syntactic, and Statistical Pattern Recognition*. Ed. by Edwin R. Hancock et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 453–462. ISBN: 978-3-642-14980-1.

[18] Hiromasa Fujihara et al. "F0 estimation method for singing voice in polyphonic audio signal based on statistical vocal model and viterbi search". In: *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*. Vol. 5. IEEE. 2006, pp. 253–256.

[19] Zoubin Ghahramani. "An introduction to hidden Markov models and Bayesian networks". In: *Hidden Markov models: applications in computer vision*. World Scientific, 2001, pp. 9–41.

[20] Marco Grimaldi, Pádraig Cunningham, and Anil Kokaram. "A Wavelet Packet Representation of Audio Signals for Music Genre Classification Using Different Ensemble and Feature Selection Techniques". In: *Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrieval*. MIR '03. Berkeley, California: Association for

Computing Machinery, 2003, 102–108. ISBN: 1581137788. DOI: 10.1145/973264.973281.

[21] Negar Hariri, Bamshad Mobasher, and Robin Burke. "Context-Aware Music Recommendation Based on Latenttopic Sequential Patterns". In: *Proceedings of the Sixth ACM Conference on Recommender Systems*. RecSys '12. Dublin, Ireland: Association for Computing Machinery, 2012, 131–138. ISBN: 9781450312707. DOI: 10.1145/2365952.2365979.

[22] Negar Hariri, Bamshad Mobasher, and Robin Burke. "Personalized text-based music retrieval". In: *Workshops at the Twenty-Seventh AAAI Conference on Artificial Intelligence*. 2013.

[23] Yin-Fu Huang et al. "Editorial: Music Genre Classification Based on Local Feature Selection Using a Self-Adaptive Harmony Search Algorithm". In: *Data Knowl. Eng.* 92 (July 2014), 60–76. ISSN: 0169-023X. DOI: 10.1016/j.datak.2014.07.005.

[24] Dietmar Jannach, Lukas Lerche, and Iman Kamehkhosh. "Beyond "Hitting the Hits": Generating Coherent Music Playlist Continuations with the Right Tracks". In: *Proceedings of the 9th ACM Conference on Recommender Systems*. RecSys '15. Vienna, Austria: Association for Computing Machinery, 2015, 187–194. ISBN: 9781450336925. DOI: 10.1145/2792838.2800182.

[25] Jesper H. Jensen et al. "Quantitative Analysis of a Common Audio Similarity Measure". In: *IEEE Transactions on Audio, Speech, and Language Processing* 17.4 (2009), pp. 693–703. ISSN: 1558-7924. DOI: 10.1109/TASL.2008.2012314.

[26] Guolin Ke et al. "LightGBM: A Highly Efficient Gradient Boosting Decision Tree". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 3146–3154.

URL: http://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree.pdf.

[27] Eamonn Keogh. "Exact Indexing of Dynamic Time Warping". In: *Proceedings of the 28th International Conference on Very Large Data Bases*. VLDB '02. Hong Kong, China: VLDB Endowment, 2002, 406–417.

[28] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: http://arxiv.org/abs/1412.6980.

[29] Tetsuro Kitahara. "Mid-level Representations of Musical Audio Signals for Music Information Retrieval". In: *Advances in Music Information Retrieval*. Ed. by Zbigniew W. Raś and Alicja A. Wieczorkowska. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 65–91. ISBN: 978-3-642-11674-2. DOI: 10.1007/978-3-642-11674-2_4.

[30] Anssi P. Klapuri. "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness". In: *IEEE Transactions on Speech and Audio Processing* 11.6 (2003), pp. 804–816.

[31] Peter Knees et al. "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*. MIR '06. Santa Barbara, California, USA: Association for Computing Machinery, 2006, 147–154. ISBN: 1595934952. DOI: 10.1145/1178677.1178699.

[32] Paul B. Lamere. "I've Got 10 Million Songs in My Pocket: Now What?" In: *Proceedings of the Sixth ACM Conference on Recommender Systems*. RecSys '12. Dublin, Ireland: Association for Computing Machinery, 2012, 207–208. ISBN: 9781450312707. DOI: 10.1145/2365952.2365994.

[33] Chang-Hsing Lee et al. "Automatic Music Genre Classification Based on Modulation Spectral Analysis of Spectral and Cepstral Features". In: *IEEE Transactions on Multimedia* 11.4 (2009), pp. 670–682.

[34] Arto Lehtiniemi and Jarno Seppänen. "Evaluation of Automatic Mobile Playlist Generator". In: *Proceedings of the 4th International Conference on Mobile Technology, Applications, and Systems and the 1st International Symposium on Computer Human Interaction in Mobile Technology*. Mobility '07. Singapore: Association for Computing Machinery, 2007, 452–459. ISBN: 9781595938190. DOI: 10.1145/1378063.1378135.

[35] Blerina Lika, Kostas Kolomvatsos, and Stathes Hadjiefthymiades. "Facing the cold start problem in recommender systems". In: *Expert Systems with Applications* 41.4, Part 2 (2014), pp. 2065 –2073. ISSN: 0957-4174. URL: http://www.sciencedirect.com/science/article/pii/S0957417413007240.

[36] D-Lib Magazine. "The Music Information Retrieval Evaluation eXchange (MIREX)". In: *D-Lib Magazine* 12.12 (2006), pp. 1082–9873.

[37] Michael I. Mandel and Dan Ellis. "Song-Level Features and Support Vector Machines for Music Classification". In: *ISMIR 2005, 6th International Conference on Music Information Retrieval, London, UK, 11-15 September 2005, Proceedings*. 2005, pp. 594–599. URL: http://ismir2005.ismir.net/proceedings/1106.pdf.

[38] Brian McFee and Gert RG Lanckriet. "Hypergraph Models of Playlist Dialects". In: *Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012, Mosteiro S.Bento Da Vit 'o ria, Porto, Portugal, October 8-12, 2012*. Ed. by Fabien Gouyon et al. FEUP Edi c c o es, 2012, pp. 343–348.

[39] Brian McFee et al. "The million song dataset challenge". In: *Proceedings of the 21st International Conference on World Wide Web*. 2012, pp. 909–916.

[40] Hugh McIntyre. *The Top 10 Streaming Music Services By Number Of Users*. Accessed on 3 September 2020. 25 May 2008. URL: https://www.forbes.com/sites/hughmcintyre/2018/05/25/the-top-10-streaming-music-services-by-number-of-users.

[41] Emilia G ’o mez and Perfecto Herrera. “Estimating The Tonality Of Polyphonic Audio Files: Cognitive Versus Machine Learning Modeling Strategies”. In: *ISMIR 2004, 5th International Conference on Music Information Retrieval, Barcelona, Spain, October 10-14, 2004, Proceedings*. 2004. URL: http://ismir2004.ismir.net/proceedings/p018-page-92-paper164.pdf.

[42] Y. V. Srinivasa Murthy and Shashidhar G. Koolagudi. “Content-Based Music Information Retrieval (CB-MIR) and Its Applications toward the Music Industry: A Review”. In: *ACM Comput. Surv.* 51.3 (June 2018). ISSN: 0360-0300. DOI: 10.1145/3177849.

[43] Aaron Ng and Rishabh Mehrotra. “Investigating the Impact of Audio States & Transitions for Track Sequencing in Music Streaming Sessions”. In: *14th ACM Conference on Recommender Systems*. RecSys ’20. Virtual Event, Brazil: Association for Computing Machinery, 2020. DOI: 10.1145/2792838.2800182.

[44] Tien T. Nguyen et al. “Exploring the Filter Bubble: The Effect of Using Recommender Systems on Content Diversity”. In: *Proceedings of the 23rd International Conference on World Wide Web*. WWW ’14. Seoul, Korea: Association for Computing Machinery, 2014, 677–686. ISBN: 9781450327442. DOI: 10.1145/2566486.2568012.

[45] Henri J. Nussbaumer. “The Fast Fourier Transform”. In: *Fast Fourier Transform and Convolution Algorithms*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1981, pp. 80–111. ISBN: 978-3-662-00551-4. DOI: 10.1007/978-3-662-00551-4_4.

[46] Elias Pampalk, Tim Pohle, and Gerhard Widmer. "Dynamic Playlist Generation Based on Skipping Behavior". In: *ISMIR 2005, 6th International Conference on Music Information Retrieval, London, UK, 11-15 September 2005, Proceedings*. 2005, pp. 634–637. URL: http://ismir2005.ismir.net/proceedings/2072.pdf.

[47] Sasank Reddy and Jeff Mascia. "Lifetrak: Music in Tune with Your Life". In: *Proceedings of the 1st ACM International Workshop on Human-Centered Multimedia*. HCM '06. Santa Barbara, California, USA: Association for Computing Machinery, 2006, 25–34. ISBN: 1595935002. DOI: 10.1145/1178745.1178754.

[48] Jeremy Reed and Chin-Hui Lee. "On the importance of modeling temporal information in music tag annotation". In: *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. 2009, pp. 1873–1876.

[49] J. Ben Schafer et al. "Collaborative Filtering Recommender Systems". In: *The Adaptive Web: Methods and Strategies of Web Personalization*. Ed. by Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 291–324. ISBN: 978-3-540-72079-9. DOI: 10.1007/978-3-540-72079-9_9.

[50] Christian Schörkhuber and Anssi Klapuri. "Constant-Q transform toolbox for music processing". In: Jan. 2010, pp. 3–64.

[51] Malcolm Slaney. "Web-Scale Multimedia Analysis: Does Content Matter?" In: *IEEE MultiMedia* 18.2 (2011), pp. 12–15.

[52] Tero Tolonen and Matti Karjalainen. "A computationally efficient multipitch analysis model". In: *IEEE Transactions on Speech and Audio Processing* 8.6 (2000), pp. 708–716.

[53] George Tzanetakis and Perry Cook. "Musical genre classification of audio signals". In: *IEEE Transactions on Speech and Audio Processing* 10.5 (2002), pp. 293–302.

[54] Fabio Vignoli and Steffen Pauws. "A Music Retrieval System Based on User Driven Similarity and Its Evaluation". In: *ISMIR 2005, 6th International Conference on Music Information Retrieval, London, UK, 11-15 September 2005, Proceedings*. 2005, pp. 272–279. URL: http://ismir2005.ismir.net/proceedings/1021.pdf.

[55] Manzil Zaheer et al. "Deep Sets". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 3391–3401. URL: http://papers.nips.cc/paper/6931-deep-sets.pdf.

[56] Elena Zheleva et al. "Statistical Models of Music-Listening Sessions in Social Media". In: *Proceedings of the 19th International Conference on World Wide Web*. WWW '10. Raleigh, North Carolina, USA: Association for Computing Machinery, 2010, 1019–1028. ISBN: 9781605587998. DOI: 10.1145/1772690.1772794.