TÉCNICO LISBOA

TASTES: A Taxonomy-Free Music Recommendation Strategy for Interest Expansion on Spotify

José Maria Moraes Pessanha de Mendonça Gouvêa

Thesis to obtain the Master of Science Degree in

Computer Science and Engineering

Supervisor: Prof. Carlos Martinho

June 2024

Declaration
I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

I would like to thank my parents and grandmother for their support, sacrifice, encouragement, faith and care over all these years, for always being there for me when I needed strength or just a helping hand and without whom I would have never been in the position to develop this project.

I would also like to acknowledge my dissertation supervisor Prof. Carlos Martinho for his insight, support, availability and patience, without his belief in my vision and his aid in developing this project this Thesis would not have been possible.

Last but not least, to all my friends, at home and abroad, without whose presence, joy and support I would not have been able to accomplish this endeavour. And to Molly, for being a bright light even in my darkest days. Thank you.

To each and every one of you – Thank you.

Abstract

With the continuous rise of digital music streaming platforms, the demand for more adaptive and innovative music recommendation systems has grown exponentially. *Spotify*, a global leader in this space, continuously strives to provide dynamic functionalities which connect millions of listeners to the right musical content. Aiming to explore new ways of enhancing music discovery in these platforms, we introduce TASTES, a novel sequential recommendation approach for playlist continuation tasks on *Spotify*. TASTES is an interest expansion tool free of genre taxonomy that leverages a dynamic multifaceted model of the user's inclinations across prevailing musical categories in their hand-curated playlists, following the user's evolving trends into unexplored frontiers of their musical taste. The strategy employs *Spotify*'s Web API services for direct integration with the application, leveraging its features to enhance the user's discovery experience. To evaluate our proposed approach, we conducted a 30-day within-subjects study with 20 participants, comparing TASTES' aptitude for playlist continuation and interest expansion against a baseline strategy employing *Spotify*'s recommendation method. The strategies were evaluated according to the accuracy and diversity of the suggestions produced and the overall user experience provided, through both quantitative metrics and user-informed assessment. The study results show that TASTES does not demonstrate relevant improvement over *Spotify*'s approach as a musical interest expansion tool. However, the strategy proved more capable than its counterpart at generating relevant suggestions within the user's preferences and providing a more satisfying music discovery experience, in cases where speech level and content are less relevant factors to the listener.

Keywords

Music Recommendation; Music Discovery; Musical Interest Expansion; Playlist Continuation; Spotify; Taxonomy-Free Classification; Interest-Aware Recommendation; Sequential Recommendation.

Resumo

Com o aumento do uso de plataformas de streaming de música digital, a procura por sistemas de recomendação de música inovadores tem crescido exponencialmente. O *Spotify*, um líder global nesta arena, tenta continuamente oferecer funcionalidades dinâmicas que conectam milhões de ouvintes ao conteúdo musical certo. Com o objetivo de explorar novas abordagens para descoverta de música, apresentamos TASTES, uma nova abordagem de recomendação sequencial para continuação de playlists no *Spotify*. TASTES, uma ferramenta de expansão de interesses livre de taxonomia de géneros, utiliza um modelo multifacetado e dinâmico das inclinações do utilizador em uma das suas playlists, seguindo as suas tendências evolutivas para novas fronteiras de interesse musical. A estratégia usa os serviços da Web API do *Spotify* para integração direta com a aplicação, aproveitando suas funcionalidades para melhorar a experiência de descoberta do utilizador. Conduzimos um estudo de 30 dias com 20 participantes, comparando a aptidão de TASTES para continuação de playlists e expansão de interesses com a estratégia de recomendação do *Spotify*. Avaliamos as estratégias quanto à precisão e diversidade das sugestões, bem como à experiência fornecida ao utilizador, através de métricas quantitativas e avaliação dos utilizadores. Os resultados mostram que TASTES não demonstra uma melhoria relevante sobre a método de recomendaç ão do *Spotify* como ferramenta de expansão de interesse musical, mas é mais eficaz a gerar sugestões relevantes e proporcionar uma experiência de descoberta mais satisfatória quando o nível e conteúdo lírico são fatores menos relevantes para o ouvinte.

Palavras Chave

Recomendação de Música; Descoberta Musical; Expansão de Interesse Musical; Continuação de Playlists; Spotify; Classificação sem Taxonomia; Recomendação *Interest-Aware*; Recomendação Sequencial.

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

API	Application Program Interface
APC	Automatic Playlist Continuation
IDE	Integrated Development Environment
ILD	Intra-List Diversity
KLD	Kullback-Leibler Divergence
KNN	K-Nearest Neighbors
NDCG	Normalized Discounted Cumulative Gain
PCA	Principle Component Analysis
PC	Playlist Continuation
RFC	Recommendation-Feedback Cycle
SPAD	Subprofile-Aware Recommendation Diversification
SQL	Structured Query Language
SRSE	Spotify Recommendation Strategy Emulator
SVM	Support Vector Machines
SWAPI	Spotify Web Application Program Interface
TASTES	Trend-Adaptive Strategy for Tracklist Expansion on Spotify
TES	TASTES Execution Step
TUDB	TASTES User Database
UI	User Interface

1

Introduction

Contents

1.1 Motivation

As global music streaming platforms continue to thrive and gain popularity, the average listener's access to extensive music libraries has become an established reality. Given the wide variety of content made available through these platforms, there is a rising need for functionalities that tailor the most relevant music according to the listener's necessities and preferences. Moreover, the multifaceted and fluid nature of modern music listening habits creates an ever-growing demand for tools which introduce listeners to multiple new genres of interest, enhancing their music discovery experiences and helping expand their song collections.

The wide spectrum of research conducted into what drives satisfaction in recommendation tasks has continuously confirmed that the accuracy of suggestions is only one factor in the process, and that the novelty and diversity of recommended items are equally critical aspects in fulfilling the consumer's needs. Accordingly, one of the key challenges in music discovery lies in providing listeners with suggestions that develop their musical repertoire based on their evolving trends, thereby meeting their desire for exploration.

Spotify has become the go-to platform for millions of music enthusiasts around the globe due to its vast music library, user-friendly interface and broad variety of powerful tools for recommendation and discovery. Amongst these features, "Discover Weekly" stands out as one of the most popular, curating a list of new music tracks every week which lie outside of the user's usual interests. The feature serves as a successful example of how expansion-driven functionalities can enhance music discovery tasks. It highlights the importance of following the user's preferences and venturing into unexplored musical genres and styles.

Spotify also offers functionalities to help users organise their libraries and build their music collections. One prominent feature is the ability to curate playlists, allowing users to hand-select tracks to assemble tailored collections with an overarching theme and/or desired purpose. These playlists often reflect the user's perception and categorisation of their own musical inclinations, where facets of their interests are self-defined through groupings of tracks bound by a theme or set of characteristics known only to the playlist's creator.

Hand-curated playlists have become a favoured method for organising, listening and sharing different types of music collections among *Spotify* users. Accordingly, the streaming platform provides some functionalities designed to suggest new songs which the user might be interested in adding to a specific playlist. Although these recommendation features effectively cater to the themes and genres included in the user's playlist, none has the aim of expanding the musical spectrum represented in them.

A commonly discussed limitation in traditional music discovery approaches is the reliance on predefined genre taxonomies. Genre classifications can be restrictive and often fail to capture the multidimensional quality of musical trends. This limitation becomes more evident when considering the increasing amount of contemporary music which does not conform to traditional genre categories, often lying at the boundaries of musical themes and/or incorporating elements from various styles. Furthermore, different tracks within the same category can vary significantly in style, mood, or instrumentation, hindering a taxonomy-dependent system's ability to offer a more personalised music discovery experience, capable of capturing and adjusting to the user's unique and evolving interests.

1.2 Approach and Research Questions

To tackle the problems presented, we propose a novel interest-aware recommendation approach: Trend-Adaptive Strategy for Tracklist Expansion on Spotify (TASTES). The primary objective of the strategy is to expand the user's musical interests, represented in one of their hand-curated playlists, by generating personalised suggestions for the continuation of that playlist which follow the user's evolving trends across the different music categories represented, guiding them into new horizons of music discovery.

A key characteristic of TASTES is its taxonomy-free approach to music classification. Through unsupervised clustering techniques, the strategy identifies congruent groupings of songs within a playlist without relying on predefined genre taxonomies. This flexibility allows TASTES to adjust to different song collections and provide a more adaptive categorisation of tracks.

Considering our motivation in this project, we define our two primary research questions as follows:

1. How effective is a taxonomy-free interest-aware recommendation approach such as TASTES at expanding the listener's musical interests?
2. Can a taste expansion approach to recommendation improve listeners' music discovery experience?

To assess the effectiveness of our approach in light of the two research questions, we conduct a user study using a practical implementation of TASTES, integrated into the *Spotify* application. Our methodology will include evaluating the accuracy, diversity, and user satisfaction with the generated suggestions compared to a baseline strategy employing *Spotify*'s recommendation framework by leveraging the services offered through their Web API. By analysing the impact of our strategy on music discovery and its ability to expand the user's interests within the context of one of their hand-curated playlists on *Spotify*, we aim to provide valuable insights into the role of expansion-driven approaches for recommendation in modern conditions.

1.3 Contributions

By means of our research, we make three main contributions to the field of music discovery:

- We conduct a comprehensive literature review, exploring the existing research on music recommendation systems, diversity, and the challenges of genre taxonomy in music classification. This review provides a solid foundation for our proposed approach, highlighting relevant aspects which informed our design.
- We develop a novel music recommendation model, TASTES, that focuses on expanding the user's musical interests within the context of their hand-curated playlists. By leveraging an evolving model

of their preferences across the playlist's sub-categories, the strategy generates personalised recommendations that extend the user's inclinations into new areas of interest. The strategy aims to enhance the user's music exploration journey through a sequential recommendation framework, free of a predefined genre taxonomy.

- We conduct a user study to gain further insights into the effectiveness of the proposed strategy for music discovery on *Spotify*. We assess the impact of employing our approach for playlist continuation tasks from both a quantitative and user-centred perspective. The results from this assessment contribute to a better understanding of expansion-driven functionalities' potential for enhancing modern recommendation tasks.

1.4 Outline

This document is structured as follows:

- **Section 2: Related Work** outlines all the relevant research in the fields of music classification and interest-aware recommendation which informed our implementation and evaluation, introducing some important concepts which underlie the language and mechanisms presented in this paper.
- **Section 3: Implementation** describes the architecture behind our practical implementation of TASTES, outlining the core aspects of our approach, its structural components, and processing pipeline, as well as introducing the devised baseline strategy for evaluation.
- **Section 4: Evaluation** describes the procedure taken in our user study assessment of TASTES, detailing the experimental settings, evaluation metrics and data collection methods employed to assess the primary research questions.
- **Section 5: Results** presents and examines the results of the user study, describing the participant sample, the evaluation methodologies used and their corresponding findings in light of the research questions posed.
- **Section 6: Conclusion** presents the final conclusions drawn from the totality of the research presented in this paper, summarising the key findings, limitations encountered, and future directions for further investigation and assessment.

2

Related Work

Contents

In this chapter, we delve into the foundational methodologies and innovative frameworks underpinning our research. We begin by introducing some fundamental aspects which outline the particular challenges we tackle in the music recommendation landscape. Additionally, we detail the tools and mechanisms which guided and enabled our *Spotify* integration method. Following this, we dissect the prior research work which provided the basis for this project. We examine taxonomy-free music classification, highlighting its significance in enhancing recommendation systems, before moving on to interest-aware recommendation strategies and their potential for playlist continuation by balancing accuracy and diversity in suggestions. Finally, we summarise the key findings of our research in light of the design and approach taken in our proposed solution.

2.1 Fundamental Concepts

Before delving into the literary review which provided the technical foundation for the conceptualisation and practical implementation of our approach, we first introduce some of the vital concepts to the topics explored in this paper. We begin by describing the underlying notions defining our project's area of focus and specific challenges tackled within the scope of preference elicitation and recommendation. Subsequently, we provide a brief overview of Spotify Web Application Program Interface (SWAPI), and the functionalities leveraged in our solution which allowed its integration into *Spotify* profiles.

2.1.1 Recommendation

To better contextualise our approach and understand its place within the broader framework of music recommendation strategies, we describe some of the basic concepts that inform our methodology and then progress into the more specific realm of challenges we tackle in this project.

2.1.1.A Recommender Systems

At their most fundamental state, recommender systems are intelligent algorithms[1] designed to suggest items, such as books, movies, or songs, that a user might be interested in. These systems aim to tackle preference elicitation tasks where they must predict a user's interest or affinity for an item based on any available feedback information, such as their past behaviour and interactions, explicitly stated preferences, or even the preferences shown by similar users. They have been widely adopted across a range of digital platforms to enhance user experience and personalise content delivery [1].

2.1.1.B Diversity-Aware Recommendation

The importance of diversity in recommendation has long been an established topic of research, spawning a wide variety of different approaches and practical implementations [2] [3] [4]. A diverse set of recommendations can engage the user more, inspire exploration, and prevent the homogenisation of content [5]. Diversity-aware recommendation approaches aim to maximise the variety of items presented to a user. Their process is rooted in the understanding that user interests are multi-faceted and evolve over time, necessitating a range of suggestions that cover the broad scope of their potential interests. The importance of diversity-aware recommendation systems becomes even more evident in contexts where user preferences are multifaceted, ambiguous or transitory, such as music. A music listener might enjoy multiple combinations of genres and artists, without following any clear-cut definition, and their preferences can shift dynamically based on real-life context, mood or socio-economic

[1] An algorithm is a precise step-by-step plan for a computational procedure that often begins with an input value and yields an output value, in a finite number of operations.

environment [6]. An approach which incorporates diversity in recommendations will be more capable of capturing the user's different interests and therefore have a better understanding of their changing needs.

2.1.1.C Interest-Aware Recommendation

Interest-aware, or intent-aware, recommendation strategies are a sub-category of diversity-aware systems. These approaches not only aim to maximise the variety of the recommendations produced but also align this variety closely with the user's relative preferences towards their interests. By tracking where the user's tendencies lie within their different categories of interest, the strategies attempt to produce suggestion sets where each category is represented according to the user's relative affinity towards it. They engage in a controlled variety framework which incorporates the objectives of a diversity-driven methodology while offering more personalisation and adaptability to the user's multifaceted needs.

In the context of music recommendation, adopting an interest-aware recommendation approach holds several advantages. First, it acknowledges the complexity and dynamism of musical tastes, which are rarely static and often cut across traditional genre boundaries. Secondly, it offers a more malleable framework that capable of readjusting to the user's fluid trends, allowing for a more relevant and engaging recommendation experience. In the next section of this chapter, we review two interest-aware strategies that tackle the task of music recommendation with proven efficacy.

2.1.1.D Playlist Continuation (PC)

With the rise in popularity of online streaming platforms like *Spotify*, which promote and facilitate content organisation through playlists, PC has become a commonly studied challenge in the realm of music recommendation. In a PC problem, the goal is to produce song suggestions to be added to one of the user's playlists, based on their compatibility with both the tracklist's musical footprint and the user's inclinations at the time they are requested. Thus, the problem does not only test a recommendation methodology's ability to produce adequate candidates for continuation but also its capacity to cater to the user's changing preferences within the interests represented by their playlist. This particularity made the task an ideal candidate for the experimental setting used in TASTES' evaluation.

2.1.2 Spotify

Spotify is a leading music streaming service that offers access to an extensive library of music and podcasts. The platform is designed to cater to individual tastes while also promoting music discovery and cataloguing, making it a central hub for audio entertainment all around the world. We now introduce

the key features and functionalities necessary to understand the integration of our proposed approach with the *Spotify* platform, along with the fundamental concepts underlying this integration.

2.1.2.A *Spotify Library*

The *Spotify Library* is a personalised collection where users can save their favourite music and podcasts. It includes albums, artists, playlists, and tracks that users have chosen to keep for easy access and streaming. This library lets users organise and quickly find their preferred content, creating a central place for everything they enjoy listening to on *Spotify*.

2.1.2.B Playlist Continuation Features

The *Spotify* application currently offers users four distinct features for the task of PC:

- ”Recommended Songs”: A short list of tracks, presented to the user at the bottom of any personal playlist, meant as suggestions for the user to add to the playlist.

- ”Enhanced Shuffle”: A shuffle play option variant for playlists which inserts new tracks in between songs in the listening queue, so that it alternates between user tracks and *Spotify* suggestion for continuation, offering a hybrid music discovery experience.

- ”Automatic Playback Continuation”: If a user is listening to a playlist’s tracks successively in a single session and reaches the end of the list, *Spotify* automatically inserts new continuation suggestions in the user’s listening queue to maintain their ongoing streak until they finally wish to interrupt it.

- ”Create a Similar Playlist”: This feature generates a new playlist composed of tracks similar to the ones in a playlist chosen by the user. This feature is only available on *Spotify*’s desktop application.

Given *Spotify* does not disclose any information regarding the different recommendation techniques behind each of the aforementioned features, we cannot derive any conclusion about the specific way the tracks are selected. Nevertheless, even though these functionalities may provide effective forms of continuing the user’s playlists, they are not designed to inherently promote the discovery of new music styles, which is a significant shortcoming for users seeking to diversify their tracklists. The need for this kind of exploratory experience is accentuated by one of *Spotify*’s most successful functionalities, ”Discover Weekly” [7], which curates the best candidate songs the user might enjoy, outside of their usual preferences.

2.1.2.C Spotify Web Application Program Interface (SWAPI)

SWAPI [8] is a powerful RESTful Application Program Interface (API)[2] that gives independent developers access to a wide range of functionalities for *Spotify* music and profile data collection, analysis and publishing. The API serves as a bridge between third-party applications and the *Spotify* database, allowing external applications to retrieve information about tracks, albums, artists and playlists, as well as edit the users' *Spotify Libraries*. The data handling tool has been extensively employed for a wide variety of applications, including programs for music analysis [9], playlist curation instruments [10], and integrated solutions for other platforms [11]. Here we provide a brief description of the API functionalities employed in our proposed solution. We note that all the detailed functions have rate limits directly imposed by *Spotify* [12] which prevent any application from performing too many endpoint requests in a short time frame. Some of the design choices in our approach were purposefully taken to adapt to these restrictions and ensure we maximise each operation's utility for its intended purpose.

Audio Feature Requests

SWAPI offers a comprehensive audio feature analysis endpoint, allowing access to a wide range of quantitative metrics that describe a track's musical footprint. For every track on the platform, data is provided on a total of 12 distinct audio descriptors some of which are low-level traditional music metrics, such as tempo[3] and key[4], while others are high-level quantifiers, such as "energy" and "danceability", determined by *Spotify*'s own audio computation analysis. The specific processes through which the latter group of features is determined are left undescribed by *Spotify*. Nevertheless, SWAPI's documentation includes a brief summary of the audio characteristics measured by each descriptor. The names, value ranges and descriptions provided for each audio feature retained are presented in Table A.1. The audio feature endpoint accepts requests for up to 100 tracks at once. Given a list of all the required tracks' *Spotify* IDs[5], the requests return the values of all 12 features presented for each song in the list.

Playlist Requests

SWAPI offers extensive functionality for managing user-curated playlists. Amongst these is the ability to publish playlists in a user's *Spotify Library*, with the option of providing a custom title, description and cover image. After a playlist has been successfully uploaded, the publish request returns its *Spotify* ID as confirmation. Any published playlist is automatically owned by the user whose profile it is posted in, allowing them to make any changes to the tracklist and personalise the other aesthetic aspects to their liking through the application interface. Additionally, the API includes functions to both add and replace

[2] A representational state transfer interface that two computer systems use to exchange information securely over the internet, providing a flexible, lightweight way to integrate applications and connect components in microservices architectures.
[3] In music, tempo is the speed or pace of a given composition, measured as beats per minute.
[4] In music, key is a group of pitches or notes that form the harmonic foundation of a composition.
[5] The *Spotify* ID is a unique base-62 identifier for every artist, track, album and playlist included in the platform's database.

tracks from an existing playlist owned by the user, with a maximum of 100 songs per request, provided the corresponding *Spotify* ID. Lastly, the endpoint also supports playlist read requests which return a comprehensive description of all the available data pertaining to a user-owned playlist. Amongst the retrieved data are all the tracklist's *Spotify* IDs and the Unix timestamps of when each song was added to the playlist.

Recent Streaming History Requests

The API also provides an endpoint for retrieving users' recent playback history. Each request returns the most recently streamed tracks by the user at a particular point in time, determined by a provided Unix timestamp in milliseconds. The requests can return up to 50 tracks per call, with the ability to paginate through previous listening sessions and/or define a lower timestamp threshold, up until which the analysis should be performed. Every response includes some metadata pertaining to the played tracks along with the corresponding playback timestamps of each stream. Preliminary testing has revealed that this kind of SWAPI request is not only costly in computation time but also severely restricted by *Spotify* in regards to the rate of usage in short time frames.

Recommendation Requests

SWAPI incorporates a robust functionality for obtaining track suggestions through their recommendation method. Each request accepts up to five seed items, which can include individual tracks, artists, or even genres. The seeds serve as the basis for the generated recommendations, effectively shaping the musical direction of the selected tracks, decided through *Spotify*'s proprietary recommendation algorithms. The requests can return up to 100 tracks, as long as there is enough available seed data to do so. Additionally, they support audio attribute value filtering to restrict the returned songs to a minimum, maximum or specific feature value. Although this function offers more control over specification, we refrain from using these filters in our approach as previous research revealed they often jeopardise the computational efficacy and accuracy of *Spotify*'s recommendation responses.

2.2 Literature Review

In this section, we provide a brief summary of the research done in the field of music recommendation that is relevant to the topic tackled in this project. Firstly we discuss how taxonomy-free approaches can provide a more adaptive framework for music classification as opposed to techniques that are based on a predefined genre designation. Subsequently, we introduce an interest-aware recommendation strategy, which focuses on mitigating the accuracy-diversity trade-off in recommendation sets. Finally, we analyse a practical application of one of the introduced strategies for the PC task on *Spotify*.

2.2.1 Taxonomy-Free Music Classification

Music classification has been a common topic of discussion in the field of preference elicitation. In any music recommendation task, there is an intrinsic need to establish a methodology that defines the boundaries between similar and dissimilar songs. To cater to the listener's interests within the requirements elicited by a specific recommendation context, any strategy must first secure a comprehensive model of music categories. Without this model, the task of positioning the listener's preferences within the entire musical spectrum becomes virtually impossible. A music classification framework allows a recommendation approach to zero in on a particular area of music when looking for relevant suggestions, ensuring more precision and efficiency in recommendation. A strategy's categorisation model also defines its approach to elicitation, given it informs its interpretation of correlations between tracks and musical themes. Consequently, various research has been conducted into different methods of grouping similar songs according to a variety of characteristics, and the impact of employing each approach for music preference elicitation.

In Scaringella et. al.'s 2006 **"Automatic genre classification of music content: a survey"** [13], the authors provide a comprehensive review on the topic of music genre classification methodologies. The authors present the main techniques used in defining and modelling music genres, from manual annotation to automatic methods based on signal processing and machine learning.

The survey exposes several challenges faced by researchers in the field, including the frequent lack of a clear definition of musical genres, the subjectivity and fluidity of genre classification across individual listeners and real-life contexts, and the extensive range of musical styles and sub-genres which can be defined according to the partitioning framework employed. Considering all these factors, determining a universal genre taxonomy becomes an exceptionally arduous task, a point echoed by Pachet et. al.'s earlier research [14].

Scaringella et. al. put forth some crucial aspects to consider when designing a robust genre classification framework. The authors highlight the importance of accounting for different aspects of audio similarities in classification, explored by previous authors, such as melody and harmony [15], rhythm [16], and timbre [17]. They also emphasise the significance of having efficient high-level descriptors for these characteristics, to allow the manipulation of more meaningful information and reduce processing complexity.

Concerning the topic of novelty in music classification strategies, Scaringella et. al. stress the importance of recognising and accounting for outliers when generating groups of similar songs, as explored previously by Flexer et. al. [18]. They argue that it is preferable to leave tracks unlabeled than to force a classification when one cannot be clearly ascertained. Furthermore, they propose that a multi-label classification strategy could offer more malleability in measuring and modelling music, especially when

there is more overlap between genre categories.

The authors move on to outline the importance of supporting genre evolution in any classification strategy. As stated previously, genre categories are often fluid, even more so if they are extrapolated from a user's ever-evolving and highly subjective music taste. Therefore, a robust genre taxonomy should not only allow existing genres to evolve, but also be able to expand itself, in terms of width, when a new genre emerges, and depth, when a genre can be split into two distinct sub-classifications.

On classification strategies, the survey presents some commonly used supervised learning models, their found success, and limitations. The authors argue that unsupervised methods can prove useful in scenarios where there is no clear definition of genres or where the genre taxonomy is too complex or subjective, informed by results obtained in Scaringella et. al.'s earlier work [19].

Krebbers, N.D., in his 2020 thesis **"Automatic Categorization of Electronic Music Genres"** [9] analyses the automatic categorisation of electronic music genres using an unsupervised clustering method based on *Spotify* audio feature similarity. Krebbers's main objective is to investigate whether an unsupervised approach can lead to a different classification of electronic music compared to the one produced by supervised approaches, which rely on a predefined genre taxonomy. The paper argues that taxonomy-free strategies can provide a more flexible and personalised approach to genre classification, which can be more reflective of a user's true preferences.

The methodology used in the thesis involved collecting a dataset of electronic music tracks from popular *Spotify* playlists, labelled to a specific electronic music sub-genre. These labels are assigned by either users or *Spotify* themselves through the titles and descriptions of the selected playlists, whose popularity ensures the accuracy of the genre assignment. All tracks' audio features are extracted from *Spotify*'s database, through SWAPI's data retrieval functionalities. Both supervised and unsupervised clustering algorithms are tested on how they classify the tracks into different sub-genres. The resulting distributions are then evaluated according to the accuracy of the classifications.

Both supervised classification algorithms tested, Support Vector Machines (SVM) and K-Nearest Neighbors (KNN) [20], showed good classification accuracy (approx. 70%). These results imply that the common sub-genre taxonomy seems to be a good enough classifier for types of electronic music on *Spotify* when considering audio feature similarity. It's worth noting however that the supervised strategies seem to fail most when classifying songs between two similar genres. These miss-classifications seem to align with Scaringella et. al.'s conclusions about the limitations of a fixed taxonomy in fringe cases [13].

K-Means classification [21] was chosen as the unsupervised clustering method for the author's comparative analysis. The partitioning method attempts to find the optimal distribution of a collection of data points, each represented by a set of scaled values for the same attributes, into a given number of groups,

normally referred to as "*k*", relying solely on their relative positions in the attribute space. Krebbers studied the distributions produced by the clustering strategy for different values of *k*, evaluating them on both their Silhouette Score[6] [22] and a cluster purity score, proposed by C. D. Manning [23], which measures the accuracy of the classifications according to each track's previously assigned genre label. After a Principle Component Analysis (PCA)[7] [24] was performed on the unsupervised data, K-Means was able to achieve a purity of 49.7% and a Silhouette score of .283. The results produced seem to indicate that the accepted genre taxonomy cannot be solely derived from audio similarity, but is also heavily influenced by other external factors such as cultural and social impact.

Finally, Krebbers performs a side-by-side comparison of the supervised and unsupervised classification results. The approaches produce vastly different sub-genre distributions, as illustrated by Figure 2.1 and Figure 2.2. Nevertheless, some clusters produced by the supervised strategies seem to be well-defined in the unsupervised results. This could indicate some sub-genres are more closely related to audio similarity than others. The author also denotes that sparse clusters in the supervised distribution are often split or even disappear when using unsupervised detection strategies. This disparity demonstrates how unsupervised classification methods can provide new ways of interpreting and categorising genres, distinct from the established taxonomy, which may offer new grounds for more robust and personalised preference elicitation approaches.

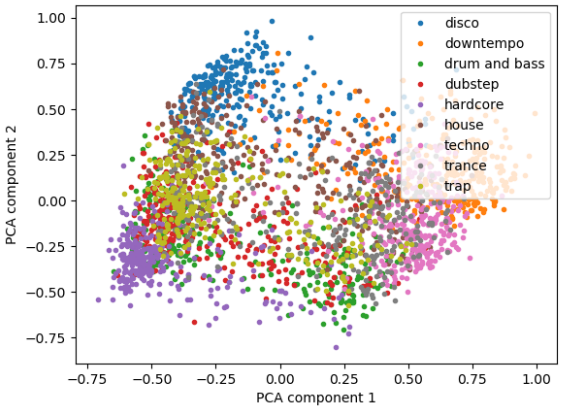


Figure 2.1: Krebber's Supervised Clustering Results

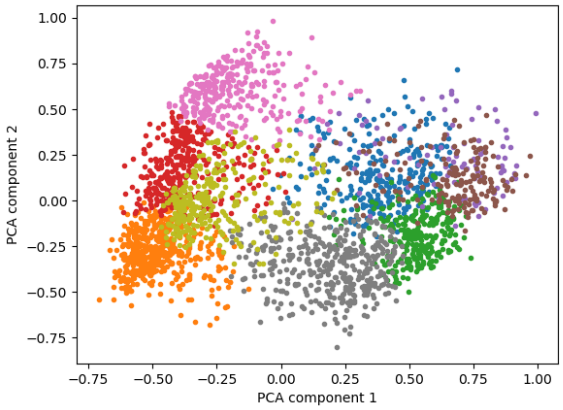


Figure 2.2: Krebber's Unsupervised Clustering Results

2.2.2 Interest-aware Recommendation

The importance of diversity in recommendation tasks has been a frequently discussed topic in the field of preference elicitation [25] [2]. Research has shown that recommendation accuracy is only one aspect of producing a satisfying user experience and that systems which take into account other factors

[6]The Silhouette Score is a measure of how similar an object is to its own cluster (cohesion) compared to other clusters (separation), ranging from -1 to 1 where a higher value indicates more distinguishable clusters.

[7]PCA is a linear dimensionality reduction technique used to reduce the number of features in a dataset while preserving the most important information.

such as the coverage and monotony of the suggested items seem to be more capable of catering to users' multifaceted needs [5]. Following this premise, an extensive range of research has been done on methodologies aimed at achieving an optimal balance between accuracy and diversity in recommendations to enhance the user's discovery experience.

Kaya and Bridge, in their 2019 **"Subprofile-aware diversification of recommendations"** [26] suggest that traditional strategies tend to focus solely on recommending popular items, which often leads to a drop in diversity and ultimately a less satisfying user experience [2]. With the purpose of better mitigating the accuracy-diversity tradeoff in sets of recommended items, the authors introduce a novel interest-aware approach which takes into account a user's multiple interests by identifying emerging groups of similar items in their consumption habits. Traditional interest-aware implementations focus on curating the suggestions that best fit the user's relative interest towards particular item traits, often referred to as aspects, reflected by a distribution of weighted values expressing the user's affinity towards each item characteristic [27]. Kaya and Bridge's method works similarly, apart from one distinction: instead of interpreting a user's preferences as a distribution of propensities across aspects, they consider the user's relative affinity towards categories of similar items they show interest for, which the authors refer to as a user's "sub-profiles". These categorisations are extrapolated from patterns in the user's consumption habits, in the interest of achieving a model which is more in line with the user's way of classifying their interests, instead of trying to impose a predetermined taxonomy that the user may not adhere to. The strategy proposed by Kaya and Bridge involves two principal stages: sub-profile identification and recommendation diversification.

Let us begin by analysing the second step of the process: recommendation diversification through Subprofile-Aware Recommendation Diversification (SPAD), an interest-aware strategy based on a previously proposed aspect-driven approach called xQuAD, introduced by Vargas in 2014 [28], and later adapted by Castells, in 2015, as a framework for interest-aware recommender systems [27]. SPAD's main goal is to generate diverse recommendations catering to the user's relative affinity for each of their sub-profiles. Similarly to xQuAD's approach, SPAD presupposes the existence of a baseline recommender system, which produces an initial pool of items for suggestion, each of them with a predicted relevance score estimating their overall alignment with the user's taste. SPAD constructs the diverse set of final recommendations based on two aspects: the average baseline relevance of the set and how well it represents the user's relative affinities towards each of their sub-profiles. In a re-ranking process, the set of final recommendations is built by iteratively selecting the best candidate item from the initial pool which maximises both aspects in the final set. The relative importance attributed to maximising each aspect is regulated by a control parameter, which can be adjusted according to the desired characteristics of the final set. A similar accuracy/diversity balancing approach is employed in our proposed solution,

although both aspects are determined differently from Kaya and Bridges' approach.

Before SPAD can generate a final set of suggestions, the strategy first requires knowledge about the user's sub-profile preference distribution. The first step in the author's proposed approach serves this exact purpose: to categorise the user's items of interest into groups displaying similar characteristics, equating to the user's sub-profiles. The authors introduce 8 distinct unsupervised classification frameworks to perform this analysis. We will focus on the method which achieved the best results when used in conjunction with SPAD: NN-1. The NN-1 clustering algorithm detects the user's sub-profiles by employing a KNN framework for the categorisation of items of interest. The algorithm works on the assumption that the set of items a user has explicitly attributed a positive appraisal to is a good representation of their main interests. NN-1 first identifies the neighbourhood subset of every item i in the user's interest set I, which includes all items in I that contain i in their k-neighborhood[8]. Subsequently, let S be the set of all neighbourhood sets found, the strategy converges by pruning any neighbourhood set which is not a subset of another set in S. After all subsets are removed, the final state of S corresponds to the user's sub-profile clusters. Kaya and Bridge define the proximity between two items as the cosine similarity between rating vectors, provided by a third-party system and assigned prior to the NN-1 analysis.

SPAD's performance was measured according to its accuracy and diversity against several other interest-aware methods, amongst them xQuAD and some of its variants. All the tested approaches utilise a re-ranking strategy for recommendations and therefore require a baseline recommender to produce the initial candidate set. The strategies' accuracy was evaluated according to the percentage of items selected for recommendation which already belonged to the user's relevant set of items. Diversity was measured using five different metrics, amongst them Intra-List Diversity (ILD) and Expected Intra-List Diversity acEILD, introduced by Ziegler et. al. [29] and Vargas and Castells [30] respectively.

Kaya and Bridge's experimental results show that, in almost every configuration tested, SPAD reaches both higher accuracy and diversity, on average, when compared to the remaining strategies. More importantly, it seems to be the least susceptible strategy to the tradeoff between recommendation accuracy and diversity. Ultimately, SPAD seems to be more precise than the remaining methods while still preserving the diversity of the generated suggestions.

In their 2021 paper **Rabbit Holes and Taste Distortion: Distribution-Aware Recommendation with Evolving Interests** [31] Zhao et. al. introduce a distribution-aware recommendation approach that improves on Kaya and Bridge's SPAD implementation [32] by accounting for dynamic changes in a user's listening habits. The objective of the proposed strategy is to mitigate two common challenges in recommendation:

[8] In KNN classification, an item's k-neighbourhood is the set of its k most similar items in the item data set.

- Rabbit Hole Effect: As a recommender system evolves and better adapts to a user's preferences, it can often over-fixate on selecting recommendations which fit the user's most prominent interests, neglecting or even outright dismissing less prominent ones. If no restrictions are imposed, it may reach a point where the user is getting exposed to a mere fraction of the available content, which can lead to a variety of undesirable outcomes such as echo chambers [33] [34] and unfair content representation [35].

- Taste Distortion Problem: Defined by the authors as the mismatch between the preference distribution of a system's suggested content and the true preference distribution of the consumer. As part of their hypothesis, they propose systems that focus on mitigating this problem will naturally produce more accurate and satisfying recommendations.

To tackle these challenges the authors introduce a Taste-enhanced Calibrated Recommendation process (*TecRec*) which employs a sequential recommendation framework to learn how the listener's musical preference shifts over time. The approach leverages feedback data to estimate the user's present trends within their sub-profiles, adjusting its selection strategy to best cater to these inclinations. We refrain from describing the exact adjustment process at this stage, given an equivalent approach is employed for our solution's architecture.

Similarly to Kaya and Bridge's approach, *TecRec* selects the final set of suggestions through an iterative post-ranking process which attempts to maximise an objective function determined by the average baseline rating of the set and how well it represents the user's preferences. To minimise taste distortion in the final set, the latter aspect of the function is given by the difference between the final set's sub-profile distribution and the user's relative preference distribution across the same categories of interest. The distance between the distributions is measured through the KLD[9] [36] between them. The set's sub-profile distribution is determined by the classification vector of each item j in the set, calculated as follows:

$$\mathbf{c}_j = \left\langle \frac{in(j, s_1)}{\sum_{s \in S} in(j, s)}, \frac{in(j, s_2)}{\sum_{g_j \in G} in(j, s)}, \cdots, \frac{in(j, s_{|S|})}{\sum_{s \in S} in(j, s)} \right\rangle \tag{2.1}$$

where S is the set of the user's sub-profiles and $in(j, s_n)$ is a binary variable which assumes the value of 1 if item j belongs to sub-profile $s_n \in S$. This multi-labelled classification framework is in line with Scaringella et. al.'s findings regarding good practices in music categorisation [13]. A similar re-ranking process is also employed in our proposed solution.

In their experiments, the authors test TecRec's effectiveness in mitigating both the problems presented against other relevant recommender systems, amongst them are xQuAD [27] and SPAD [32]. All strategies were tested according to the KLD produced by the final suggestion set. *TecRec* outperformed all other approaches in this metric, with an improvement of more than 30% over the remaining strategies,

[9] The Kullback-Leibler (KL) divergence is a measure of how one probability distribution diverges from a second, expected distribution.

showing how much better this approach performs in mitigating the taste distortion problem. *TecRec* is also tested for the accuracy of its generated recommendations using the MovieLens dataset [37], which includes user-generated item ratings employed as the ground truth for testing. In this aspect, *TecRec*'s performance is compared with CaliRec, a distribution-aware recommendation approach with calibrated genre diversity [38]. The results indicate *TecRec* outperforms its counterpart in all tested settings, reaching a peak of 60% improvement over CaliRec's recommendation accuracy.

Combining the results obtained in the experiments, we can conclude recommendation strategies such as *TecRec*, which aims to mitigate the taste distortion problem, are capable of producing recommendations that better reflect a user's true taste distribution without necessarily suffering from a drop-off in the generated recommendations' relevance.

2.2.3 Playlist Continuation

In 2018, Kaya and Bridge's **"Automatic Playlist Continuation using Subprofile-Aware Diversification"** [39] proposes the authors' solution for the ACM RecSys Challenge 2018 [40]. The challenge was organised by *Spotify*, the University of Massachusetts, and Johannes Kepler University, and focused on the task of Automatic Playlist Continuation (APC). Given a playlist of arbitrary length, each contestant would have to generate up to 500 candidate song recommendations that could be added to the playlist and still maintain its original musical footprint. The participants' submissions were evaluated using a challenge set of 10,000 user-created playlists from which various tracks had been withheld. The task involved predicting the missing tracks based on different combinations of playlist titles, descriptions and number of tracks revealed. Three metrics were used for evaluation: R-precision, Normalized Discounted Cumulative Gain (NDCG), and recommended song clicks. R-precision measures the proportion of relevant tracks among the recommended ones, considering both track and artist matches. NDCG assesses the ranking quality, rewarding higher placements of relevant tracks. Recommended song clicks is a user-centred metric which calculates the number of refreshes needed in *Spotify*'s "Recommended Songs" feature before encountering a suggested track.

Kaya and Bridge's submission consisted of an adaptation of the SPAD algorithm, previously introduced by the same authors [32], to work with *Spotify* playlists. The sub-profile detection method used was NN-1, which showed the most promising results in their previous research [26] when used in conjunction with SPAD for recommendation tasks. Among the 32 teams participating in this challenge, Kaya and Bridge's implementation came seventh in their category, showing favourable results for playlist continuation in all three evaluated metrics. This is accentuated by the fact that all metrics solely focus on measuring the accuracy of the recommendations, meaning Kaya and Bridge's diversity-aware approach, the only one of its kind in the top 10 rankings, is more than able to compete with other accuracy-driven implementations while still accounting for diversity.

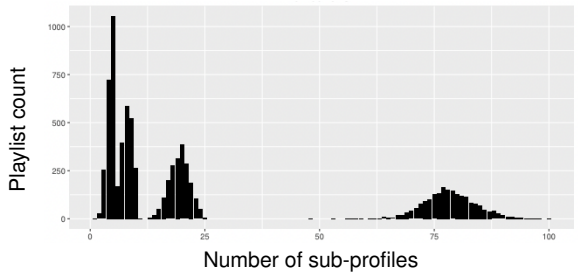


Figure 2.3: Kaya and Bridge's *Spotify* Analysis Results - Playlist Sub-profile Count Distribution

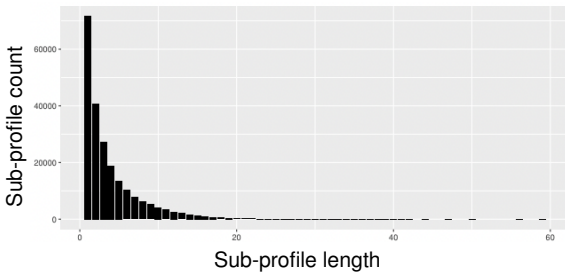


Figure 2.4: Kaya and Bridge's *Spotify* Analysis Results - Playlist Sub-profile Length Distribution

In the interest of tuning the hyper-parameters used in SPAD's re-ranking strategy, the authors analysed a set of 10000 randomly selected *Spotify* playlists for sub-profiles, using the NN-1 detection method. The results can be visualised in figures 2.3 and 2.4, which respectively show the distribution of the playlists per number of sub-profiles found and sub-profiles found according to their length, i.e. the number of songs included. The results in figure 2.3 show that the most frequent number of sub-profiles found in the analysed playlists is 4, with relative peaks at the 20 and 80 sub-profile counts. Additionally, the histogram shows that playlists with one sub-profile, i.e. playlists with no discernible musical divisions, are extremely rare. This result supports Kaya and Bridge's assertion that most *Spotify* playlists contain distinguishable sub-genres. The peaks at 20 and 80 sub-profile counts may hint at the detection of sub-partitions within the clusters, further reinforcing the idea that different classifications of music can be found at different levels of genre granularity. The histogram in figure 2.4 shows how most of the sub-profiles detected were extremely short, often composed of just one or two songs. This implies that a playlist's theme distribution can be easily altered with just a few added or removed tracks.

2.2.4 Summary of Relevant Findings

The presented literature review underscores some critical aspects which informed our solution's implementation. Firstly, we discussed the potential advantages of a taxonomy-free music classification approach in capturing the fluidity and subjectivity of musical preferences, in contrast with traditional frameworks which rely on predefined genre taxonomies. Secondly, we investigated the importance of balancing accuracy and diversity in music suggestions and the capabilities of interest-aware approaches for enhancing both the precision and variety of recommendation sets. Furthermore, we discovered how a method that adapts to dynamic changes in the user's preferences can prove effective in minimising common issues like the rabbit hole effect and taste distortion. Finally, we analysed the application of an interest-aware strategy for PC on *Spotify*, illustrating the viability of diversity-aware approaches in real-world music discovery tasks.

3

Implementation

Contents

In this chapter, we describe the methodology and architecture behind our proposed approach for *Spotify* playlist continuation: TASTES, a Trend-Adaptive Strategy for Tracklist Expansion on Spotify. We begin by outlining the core aspects behind our solution's framework for recommendation, purposefully designed to evaluate the research questions initially posed in this project. Subsequently, we provide a detailed description of TASTES's structural components and processing pipeline for recommendation production. Afterwards, we introduce a second recommendation framework, Spotify Recommendation Strategy Emulator (SRSE), developed to serve as a baseline for the evaluation of our strategy's capabilities for music discovery. Finally, we outline the technological tools used for the practical implementation of both recommendation approaches.

3.1 Methodology

Based on the particular task chosen as the experimental setting for evaluation, and the success found by the previously proposed solutions presented in Section 2.2, the following core aspects defined our proposed strategy's architectural design:

Playlist Expansion: TASTES' is specifically designed for PC tasks with the added purpose of expanding the area of musical interests represented in a playlist. Just like the music recommendation strategies introduced in the previous chapter, it requires an initial set of relevant items to the user, i.e. a hand-curated playlist, to define the area of the listener's musical interests which the approach will attempt to cater to and expand. The strategy's objective is to produce new song suggestions which follow the listener's tendencies within this area while driving them to extend its horizons by introducing music outside of the themes represented. Although TASTES can be employed for a multitude of playlists, it operates separately and independently for each one, as well as for each *Spotify* user.

Spotify Integration: Our solution's framework was purposefully designed for integration into *Spotify* profiles by leveraging the recommendation services provided through SWAPI to enhance the user's music discovery experience. *Spotify*'s Web API service is still fairly recent, and despite giving access to a wide range of different functionalities for data retrieval, it is ultimately and purposefully limited in its capacity and capabilities, as mentioned in 2.1.2.C. For this reason, some underlying design choices were taken to accommodate these limitations and maximise the utility extracted from the available functions. We further detail these particular judgments in the architectural descriptions ahead.

Taxonomy-Free Classification: Using an unsupervised classification method, the approach organises music within a playlist by finding congruent groupings of songs, referred to as the playlist's "facets", based on audio structure similarity. This framework allows the strategy to be independent of a predefined genre taxonomy, given music categories emerge as a consequence of data clustering instead of the other way around. As a result, TASTES is capable of adapting to the dataset it operates on, allowing for a more flexible way of partitioning different collections of tracks according to their audio likeness. This design aspect was inspired by the research conducted by Scaringella et. al. [13] and Krebbers, N.D. [9] and the success obtained through Kaya and Bridge's NN-1 classification method [26] [39].

Distributed Preference: TASTES is an interest-aware music recommendation approach which attempts to simultaneously cater to the listener's various interests to mitigate the accuracy-diversity tradeoff in the suggestions produced. To do so, it employs an interpretation of musical taste inspired by the diversity-aware approaches introduced in the previous chapter: a distribution of weighed affinities towards musical

styles of interest, i.e. facets in a user-curated playlist, each weight value indicating how keen the user is on listening to songs belonging to that particular category.

Sequential Recommendation: The strategy employs an iterative online framework for producing recommendations, comparable to Zhao et. al.'s *TecRec* implementation [31]. Every suggestion generation step must be triggered manually, after which the strategy performs all necessary processing in real time, concluding with the output of a new set of tailored suggestions. A model of the user's preferences within their playlist is retained across the strategy's processing iterations which we refer to as TASTES Execution Steps (TESs). The model is maintained through an external Structured Query Language (SQL) database, the TASTES User Database (TUDB), and updated at the start of each processing pipeline, based on the user's most recent preferences reflected by their *Spotify* activity. This cyclical design was specifically constructed to accommodate the rate limitations of SWAPI's data collection functionalities, through which both the user's preference data and each step's initial candidate pool are retrieved.

Evolving Interest Adjustment: Encouraged by the results achieved through Zhao et. al.'s approach to mitigating the rabbit hole effect and taste distortion problem [31], the preference distribution model employed by TASTES is also dynamic. By collecting and interpreting the most recent user preference data, inferred at the start of every TES from their *Spotify* streaming activity and interactions, the approach strives to maintain an updated representation of the state of the listener's proclivities within the playlist's facets.

3.1.1 User Feedback

TASTES relies on the user to produce new feedback data between processing iterations, pertaining to both the tracks of the playlist in focus and the suggestions previously generated by the strategy, which together define the area of interests being catered to. Processing this data allows the strategy to continuously adjust its recommendation procedure according to the listener's most recent needs and therefore select the most relevant songs to recommend at the point in time when they are requested. The approach's model of the user's preferences is updated according to three types of feedback data, all of which are collected through SWAPI endpoints:

- Recommendation feedback: Individual appraisals, assigned by the user to each TASTES recommendation, through interactions supported by the *Spotify* application interface. By attributing a "like" to a song, adding it to their playlist, removing it from the playlist where they are published or even choosing not to attribute a reaction, the user can directly communicate their level of interest in that particular type of track. We further detail each interaction and their corresponding meanings ahead.

- Playlist feedback: Modifications performed to the tracklist, reflecting the user's most recent inclinations. Songs added to the playlist are considered highly representative of the listener's tendencies within the musical spectrum in focus. Similarly, TASTES assumes any removed tracks are indicators of the types of music the user is no longer interested in listening to.

- Streaming feedback: Recent listening history is also considered when adjusting the strategy's preference model, as it provides information about the songs and playlist facets which the user most recently favours, as well as those they are lately less drawn to. The more recently and frequently a track has been listened to, the more representative of the user's present-day trends it is considered to be.

3.1.2 Radio Playlist

The recommendations produced by TASTES are directly published in the user's *Spotify Library*, in a playlist solely designated for this purpose, at the end of each TES. We refer to it as the user's **radio playlist**. The last stage in every execution step is to replace the suggestions generated in the previous iteration, still in the radio playlist, with a new set of selected tracks for the user to listen to and attribute new explicit feedback responses. Figure 3.1 shows an example of the user's *Spotify Library* view upon accessing the application, with the user and radio playlists identified accordingly. This allows the listener to easily access, identify and react to the suggestions produced by the strategy, which reanalyses the radio playlist at the start of every processing iteration to extract the necessary feedback data for the readjustment. TASTES performs all data retrieval and publishing pertaining to recommendations through SWAPI's playlist endpoints.

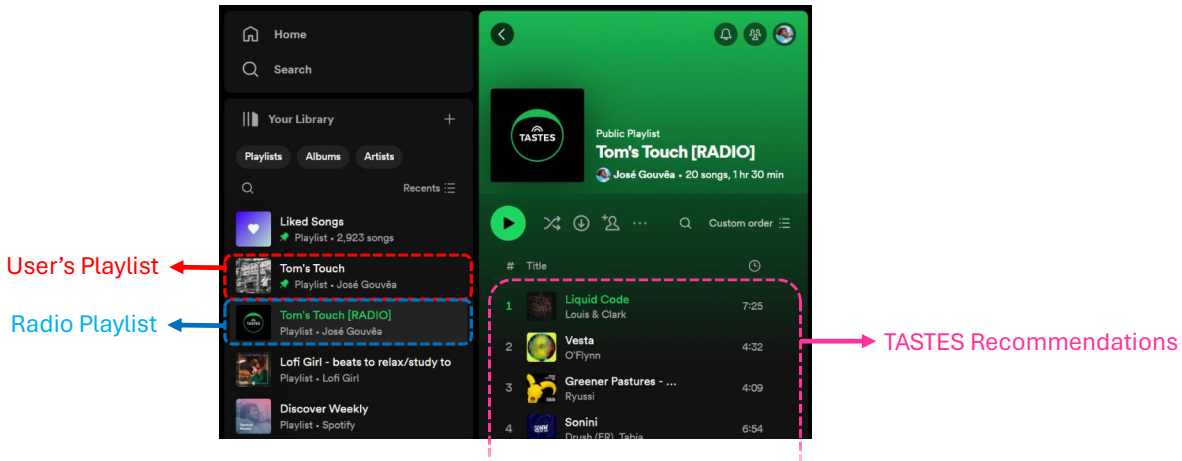


Figure 3.1: *Spotify Library* User Interface (UI) View

The strategy interprets three types of explicit feedback produced by the user in response to each suggestion published in the radio playlist:

- **Moving a track** from the radio playlist into their own reflects the user is interested in getting similar recommendations for the continuation of their playlist.
- **Removing a track** from the radio playlist indicates the user is not interested in recommendations such as that one. This interaction is equivalent to a ”dislike”, given *Spotify*’s interface does not include a mechanism for explicit negative reactions.
- **Attributing a ”like”** to a radio track indicates the user is interested in that song but does not necessarily wish to include it in their playlist.

If the user does not attribute any of the aforementioned reactions in response to a particular radio suggestion, TASTES will assume nothing regarding their interest towards the song. Furthermore, each radio track can only be assigned one of the three feedback responses interpreted by the strategy. They are superseding in the order that they are listed, meaning a song that is added to the user’s playlist will always be seen as having been attributed that reaction, even if the user ”likes” it and/or removes it from the radio playlist. The same rule applies to rejected recommendations, i.e. removed from the radio playlist, which the user has also liked.

3.2 Architecture

In this section, we detail the architecture of our proposed solution. We start by providing an overview of the data structures responsible for maintaining TASTES’ model of the user’s musical inclinations. Afterwards, we detailed the operations performed by the strategy during every first execution step TES_0, responsible for initialising the user’s preference model. Finally, we describe the mechanisms which comprise every subsequent TES’s processing pipeline. All the values of the hiper-parameters used in the TASTES’ and SRSE’s operational processes are presented in Table B.1.

3.2.1 User Preference Model

As mentioned previously, TASTES keeps track of the listener’s evolving musical taste by retaining a model of their preferences across cycles of execution. The model is updated at the start of every TES according to the most recent feedback data, extrapolated from the user’s *Spotify* profile activity. It is composed of several data structures, which collectively maintain the most recent state of the user’s inclinations within the area of interest encompassed by their playlist. At the end of every TES_i, where TES_0 corresponds to the first execution step performed over a particular playlist, the strategy retains the following updated models:

- **User Facet Preference Distribution Vector ($\mathbf{p}_i$):** An array of weights between 0-1, representing the user's updated affinity towards each facet of their playlist. Each entry $p_{i,f}$ expresses the user's relative inclination towards playlist facet $f \in F$, where F is the set of labels designating the playlist's facets. A value of $p_{i,f} = 0$ indicates the user is not interested in listening to songs belonging to the sub-category f, while 1 means the user is extremely inclined to.

- **Model Track Set Vector ($\mathbf{P}_i$):** An array of structures, designating each relevant track in the area of interests surrounding the user's playlist. Each structure contains the track's normalised audio feature values and its facet classification label. The audio feature values of song $s \in P_i$ are represented by $s_a, \forall a \in AF$, where AF is the set of twelve *Spotify* features presented in Table A.1, while the facet label is given by $s_c \in F$. The Primary Track Set Vector $P1_i \subset P_i$ contains only the model tracks included in the user's *Spotify* playlist.

- **Relevance Ratings Vector ($\mathbf{U}_i$):** An array of values between 0-1, expression the user's relative interest for each track $s \in P_i$. Each model song s's relevance rating at TES_i is given by $U_{i,s}$. A value of 0 indicates the user is not interested in the track while a value of 1 is assigned to the model song the user shows most preference for.

- **Similarity Threshold Vector ($\mathbf{D}_i$):** An array of positive values, denoting the area of similarity of each track $s \in P_i$, inside which all songs are considered akin to the s. Each entry $D_{i,s}$ retains model track s's distance to its furthest k-neighbor in P_i, computed in the audio feature dimensional space. The value of k is determined by the length of the model track set $|P_i|$ divided by the playlist's facet count $|F|$. If the audio similarity, i.e. distance, between two tracks s and r is less than s's threshold ($sim_i(s, r) < D_{i,s}$) then the strategy assumes r is significantly similar to s.

- **Audio Feature Weight Distribution Vector ($\mathbf{w}_i$):** An array of values between 0-1, expressing the user's perception of audio similarly in the area of preferences bound to their playlist. Each entry attributes weight $w_{i,a}$ expresses the importance given by the user to audio feature $a \in AF$ when including tracks in the playlist's area of interest.

- **Tabu Set ($\mathbf{TB}_i$):** An array of strings, comprised of the *Spotify* track ID's of all songs that have ever been in the user's playlist or have been recommended by TASTES for its continuation, since TES_0. Retaining the taboo set allows the strategy to avoid producing suggestions that the user has already been exposed to, as SWAPI recommendation requests do not offer a way of controlling for that aspect.

All the structures listed are maintained and updated using a third-party SQL database, referred to as TUDB. In the following subsections, we detail the mechanisms employed for initialising and updating

all data structures which comprise the strategy's user preference model, documenting the methodology and design choices behind each operation.

3.2.2 Initial Execution Step

Before TASTES can produce recommendations for a particular playlist, it must first initialise all data structures responsible for keeping track of the user's inclinations within the area of interest encompassed by their playlist. Consequently, TES_0 follows a distinct processing structure than the remaining steps, as a setup stage for the strategy's subsequent iterations over the same playlist. At this stage, due to the lack of accessible usage data before the strategy is first employed, TASTES only relies on two basic initial assumptions pertaining to the user's playlist:

- The tracklist is a good representation of the user's musical taste, at least in the context of the musical style(s) represented in the playlist.
- The more recently a track has been added to the playlist, the better it represents a user's current inclinations, at least in the context of their playlist.

Guaranteeing these aspects ensures the strategy can inform its initial analysis of the user's interests using a relevant set of items, in a similar fashion to the interest-aware strategies explored in our prior research [32] [31]. Supposing both these assumptions are true, which is likely when dealing with hand-curated *Spotify* playlists, then TASTES can initialise the user's preference model as an already decent approximation of their actual interests. Although the approach will iteratively adapt to the user's listening trends over cycles of execution, employing this initialisation framework allows it to minimise any cold-start[1] problems. This is also advantageous for our posterior performance analysis given the limitations on the time frame available for evaluation.

3.2.2.A Model Set Extraction

The first step in building the user's preference model is to assemble the initial track set vector P_0 by collecting all available *Spotify* audio feature data for each track in the user's playlist. The data is retrieved through SWAPI audio feature requests. Given the features vary in dimensional scale, all tracks' attribute values $s_a, \forall a \in AF$ are normalised to a 0-1 range for subsequent processing. At this stage, the model tracks remain without the corresponding facet classifications s_c, given the playlist's distribution must first be ascertained. An added vector T is also constructed using additional data retrieved through SWAPI playlist request where each vector entry T_s gives the timestamp of when track s was added to the user's playlist. Lastly, the primary model and tabu set vectors are also initialised as $TB_0 = P1_0 = P_0$.

[1] In computing, a cold-start refers to the problem of starting a program or operating system from a state where there is no prior knowledge or data available.

3.2.2.B Initial Relevance Rating Calculation

Once the initial model set is secured, TASTES computes the starting relevance rating vector U_0 using the previously attained added vector T. Following the strategy's initial assumptions, a model track s's initial relevance $U_{0,s}$ is determined based on how recently it has been added to the playlist. The relevance are attributed on a scale of 0-1, where $U_{0,s} \approx 0$ corresponds to the oldest song in the tracklist and $U_{0,s} \approx 1$ is assigned to most recently added one. The rating values are determined according to the following logistic function:

$$U_{0,s} = \frac{1}{1 + \exp\left(-\alpha_{recent}\left(\frac{T_s - T_{\text{oldest}}}{T_{\text{newest}} - T_{\text{oldest}}} - \beta_{recent}\right)\right)}, \quad \forall s \in P_0 \tag{3.1}$$

where T_s is the timestamp at which track s was added to the user's playlist, T_{newest} and T_{oldest} are the addition timestamps for the most recent and oldest songs in the playlist, respectively, and α_{recent} and β_{recent} represent the relevance function's slope and mid-point. A sigmoid function[2] was chosen for this calculation given it provides higher value differentiation between the old and new tracks in the playlist, compared to a linear alternative, allowing TASTES to attribute even more importance to the user's most recent inclinations. This design choice was taken after experimenting with different functions during the calibrating tests conducted prior to our evaluation, as well as with different slope and mid-point values for the same logistic function. The values used for α_{recent} and β_{recent} in our final implementation are represented in Table B.1.

3.2.2.C Audio Feature Weight Distribution Calculation

As discussed in the previous chapter, hand-curated playlists often reflect the listener's distinct way of classifying and partitioning their musical interests. To better adjust to the user's perception of audio likeness within the preference area their playlist represents, TASTES employs a weighed audio feature distance metric when determining similarities between tracks in the model. Accordingly, the similarity between two songs at TES_i is given by their weighted euclidean distance in the audio feature space, as follows:

$$sim_i(s, r) = \sqrt{\sum_{a \in AF} w_{i,a} \cdot (s_a - r_a)^2}, \quad \text{for any } s, r \in P_i \tag{3.2}$$

where $w_{i,a}$ represents the relative influence of feature a in the user's assessment of audio similarity within the model tracks, at the time of TES_i. The initial weight distribution vector w_0 is given by the inverse of each audio attribute's variance[3] in P_0 since it expresses the importance of that particular

[2] A continuous mathematical function with a distinct "S"-like curve which maps any real-valued input to a value in the 0-1 range.
[3] In statistics, variance measures the spread or dispersion of a set of data from its mean value.

characteristic in the user's grouping process:

$$w_{0,a} = \frac{1}{var_0(a)}, \quad \forall a \in AF \tag{3.3}$$

where $var_0(a)$ designates audio feature a's variance in the initial model set P_0. This approach to similarity guarantees that features with higher variation across the tracks in the model will have less influence over TASTES assessment of similarity than ones displaying more homogeneous distributions, given these are likely more relevant to the user when deciding if a track belongs in that particular area of interest.

3.2.2.D Facet Model Classification

TASTES's taxonomy-free approach to classification employs a K-Means unsupervised clustering method to identify emergent groupings of tracks in a playlist based on their audio similarity, following the method used in Krebbers, N.D.'s research [9]. Algorithm 3.1 shows the methodology used to determine a playlist's optimal facet division. As described in our breakdown of Krebbers' unsupervised classification process, the optimal partition is obtained by finding the number of k clusters, i.e. facets, which produce the distribution with the highest Silhouette Score after K-Means is performed over the tracklist's audio feature dataset $s_a, \forall a \in AF, \forall s \in P_0$. To ensure the strategy is able to represent all playlist facets in the short set of suggestions generated at the end of each TES, we imposed a maximum value for $k = MAX\_F$. A minimum facet count MIN_F was also set to avoid partitions with low cluster counts, whose value was determined based on Kaya and Bridge's sub-profile analysis presented in Section 2.2.3, since it wouldn't allow us to properly assess TASTES' interest-aware capabilities. We also note that a minimum of 20 tracks should be secured in the analysed playlist so that the strategy's facet classification process can produce relevant results. The final values used for both facet count thresholds are included in Table B.1. Finally, we note that TASTES was also designed to redetermine the playlist's facet distribution after the model set P_i has been sufficiently altered, although we did not have the chance to test this feature given the limited evaluation time frame. This design aspect was informed by Scaringella et. al.'s conclusions regarding the importance of supporting genre taxonomy expansion beyond the evolution of the classifications [13].

As illustrated in Algorithm 3.1, the clustering process employed to find the playlist's optimal partition leverages the feature weight distribution vector w_0, computed in the previous process, to determine distances between tracks in the model under the user's perception of audio similarity. Accordingly, the distance metric used in clustering is given by Equation (3.2).

Algorithm 3.1: Determine Playlist Facet Distribution

```
Input: model_track_set, feature_weight_dist, MIN_F, MAX_F
Output: facet_clusters, k_facets
max_silhouette_score ← −1
optimal_k ← 0
for k ∈ RANGE(MIN_F, MAX_F) do
    cluster_assignments ← weightedKMeans(n_clusters=k, weights=feature_weight_dist,
     data_points=model_track_set)
    silhouette_score ← getSilhouettScore(cluster_assignments)
    if silhouette_score > max_silhouette_score then
        max_silhouette_score ← silhouette_score
        optimal_k ← k
        facet_clusters← defaultdict(list)
        for track, facet_label ∈ cluster_assignments do
            facet_clusters[facet_label].append(track)
return facet_clusters, optimal_k
```

3.2.2.E Similarity Threshold Calculation

Each model track's similarity threshold $D_{i,s}$ defines the minimum distance another track must be in the model's audio feature space to be considered akin to the first, at the time of TES_i. Each value of the initial threshold vector $D_{0,s}$ is given by the distance, i.e. similarity, of each track $s \in P_0$ to its furthest k-neighbour in the initial model set, using the distance metric defined by Equation (3.2). The value of k is given by the average number of tracks per model facet $\frac{|P_0|}{|F|}$. In subsequent iteration steps, TASTES redetermines the thresholds vector D_i through the same process.

3.2.2.F Facet Preference Distribution Initialisation

Once the strategy has initialised all the necessary preference model data structures, it can finally calculate the initial estimation of the user's facet affinity distribution, represented by vector p_0. Each entry of the vector is given by the relative frequency of tracks belonging to one of the playlist's facets, as follows:

$$p_{0,f} = \frac{\sum_{s \in P_0} \delta(f, s)}{|P_0|}, \quad \forall f \in F \tag{3.4}$$

where $\delta(f, s)$ is the facet membership function:

$$\delta(f, s) = \begin{cases} 1 & \text{if } s_c = f, \\ 0 & \text{otherwise.} \end{cases} \tag{3.5}$$

Although the relative frequency of each playlist facet may not be necessarily indicative of the user's actual inclinations towards each sub-genre at the time of TES_0, it serves as a sufficiently decent estimation from which TASTES can begin its iterative adjusting procedure. From this point onward, the initial setup

step TES_0 follows the same execution stages as the ensuing processing iterations, starting from the **Recommendation Generation** stage of the subsequent pipeline description.

3.2.3 TES Pipeline

In the iterations following TES_0, TASTES' objective becomes to sequentially update the user's preference model according to the newly available *Spotify* user activity data, employing it to produce a new set of relevant recommendations for PC. Following our design's particular framework, the strategy attempts to select suggestions at the end of every iteration step which best cater to the user's most recent trends within the area of interest bound to their playlist while also pushing them towards new musical frontiers outside of the represented preferences. We now provide a description of the processes which constitute our strategy's standard iteration processing pipeline, illustrated in its entirety in Figure 3.2.

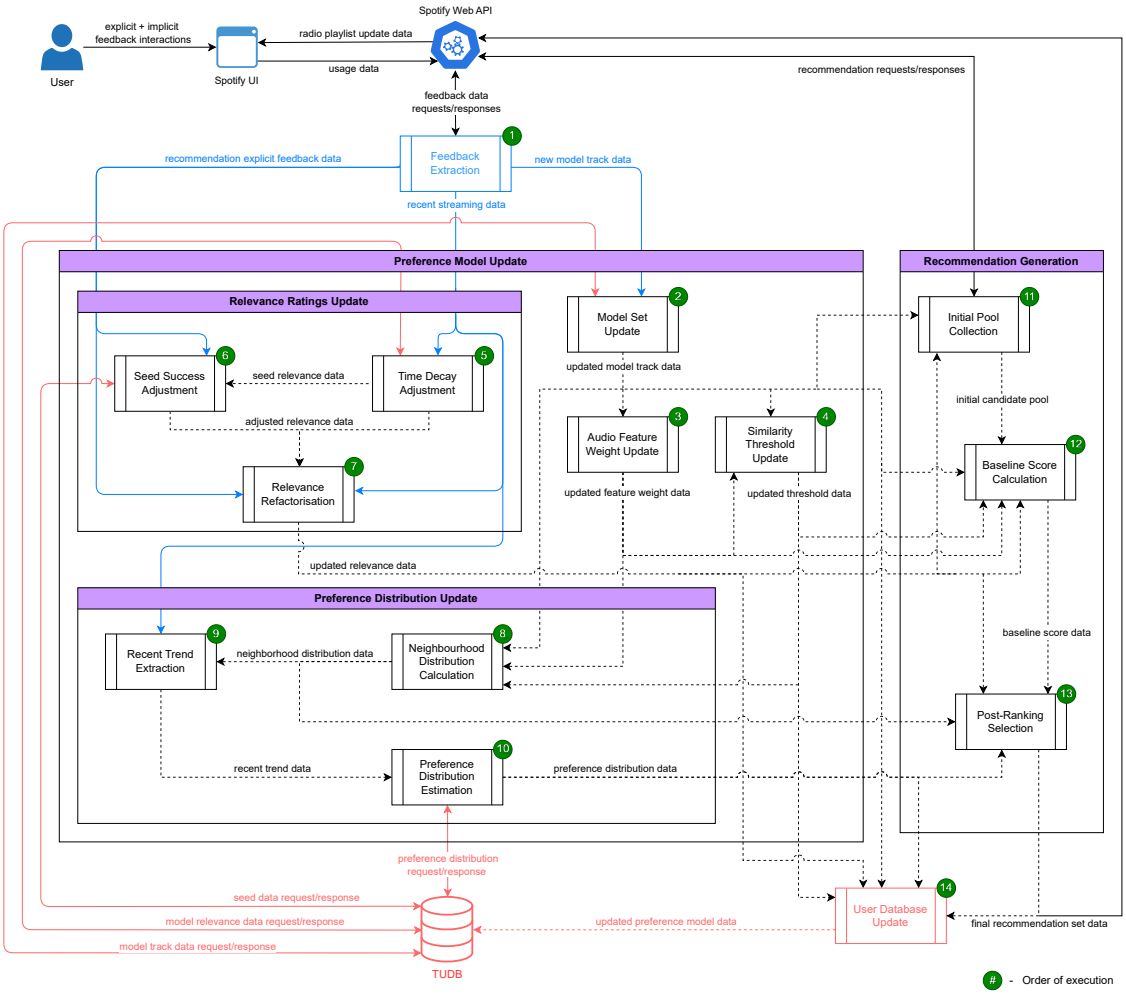


Figure 3.2: TASTES Execution Step Processing Pipeline Diagram

3.2.3.A Feedback Extraction

The first step in every TES_i is to extract all relevant feedback produced by the user since TES_{i-1}, pertaining to the three types of data interpreted by TASTES, introduced in 3.1.1:

- User Playlist Data: Extrapolate the sets of newly added and removed tracks from the user's playlist, represented by A_i and X_i respectively, by comparing the current state of the tracklist, attained through SWAPI playlist requests, with the retained primary model track set $P1_{n-1}$.
- Recommendation Interaction Data: Determine the sets of recommendations which the user has liked (RL_i), added to their playlist ($RA_i \subset A_i$), and removed from the radio playlist (RR_i). The sets are extrapolated through SWAPI playlist requests on both the user and radio playlists.
- Recent Streaming Activity Data: Generate a recent stream count vector SC_i, where every entry integer value $SC_{i,s}$ corresponds to the number of times the user has listened to a relevant track $s \in P_{i-1} \cup LR_i \cup AR_i$ since the last processing iteration was completed. The timestamps of each track's most recent stream are also retained for posterior analysis. The data is retrieved using SWAPI recent streaming history requests.

All available quantitative audio feature data is extracted and normalised for each track in sets A_i, X_i, LR_i, AR_i and RR_i, similarly to the Model Set Extraction process. In regards to recent streaming analysis, SWAPI's data collection functionalities are fairly limited. Our preliminary assessment of this particular API feature revealed two major constraints: the requests are expensive in computational time and the amount of data which can be retrieved in a short time frame is restricted by *Spotify*. For this reason, TASTES can only collect a partial representation of the intended data every iteration, producing requests for the most recent data available until the limit amount is reached.

3.2.3.B Model Set Update

After the user's feedback has been successfully collected, all new relevant songs $s \in \{A_i \cup LR_i\}$ are classified within the model's facets. Each track is attributed the most frequent facet classification $f \in F$ in their k-neighbourhood within the model set P_{i-1}, where the value of k is the same one used for the Similarity Threshold Calculation. Subsequently, the model track set is updated with the newly classified songs $P_i = \{P_{i-1} \cup A_i \cup LR_i\} \setminus X_i$, excluding the removed set. Given the model's facets are defined by the tracks which comprise them, by adding new relevant songs to P_i at every step, TASTES allows the facets to evolve and reshape, following the aspects outlined in Scaringella et. al.'s research [13]. All liked recommendations are added to the model even though the user has not decided to include them in their playlist. This is because they offer added context to where the listener's inclinations might lie outside of the themes included in the playlist, thereby better informing TASTES' interest expansion objectives.

Finally, the taboo set is also updated with just the newly added playlist tracks $TB_i = TB_{i-1} \cup A_i$, given the previous recommendation set was already included at the end of TES_{i-1}.

3.2.3.C Similarity Thresholds and Feature Weight Update

Once the model track set is updated, the strategy redetermines the audio feature weight distribution vector w_i using the same process detailed in 3.2.2.C. With the resulting adjustments to the audio similarity metric employed, TASTES will recalculate the similarity thresholds for all the tracks in the updated model, following the process described in the Similarity Threshold Calculation, resulting in D_i. The new tracks' relevance ratings $U_{i,s}, \forall s \in \{A_i \cup LR_i\}$ are only determined in the next step of the processing pipeline.

3.2.3.D Relevance Ratings Update

Once the user preference model has been updated, TASTES next step is to readjust the relevance ratings of each model track U_{i-1} according to the user's most recent inclinations, expressed by the newly collected *Spotify* activity data, and the time passed since the previous update.

During the preliminary tests conducted to assess our solution's functionality, this stage of the pipeline was merely contrived of the time decay adjustment operation. The subsequent relevance update processes were initially excluded from TASTES operational framework so that we could ascertain the values of some critical balancing factors used in their operations. We further clarify this process in the corresponding factor descriptions ahead.

Time Decay Adjustment

Firstly, the strategy starts by attenuating all relevances for the tracks in P_{i-1} based on the time elapsed since the last processing step, as follows:

$$U_{i,s} = [U_{i-1,s} \times (1 + \eta_{stream} \times \frac{SC_{i,s}}{days\_elapsed_i(s)})] \times {\phi_{decay}}^{days\_elapsed_i(s)}, \quad \forall s \in P_{i-1} \tag{3.6}$$

where SC_i is the recent stream count vector extracted in the previous pipeline process and η_{stream} and ϕ_{decay} respectively express the stream count and decay factors. If $SC_{i,s} > 0$ is true, the days elapsed function $days\_elapsed_i(s)$ gives the fraction of days passed since track s was last streamed, else it returns the same fraction for the time passed since TES_{i-1}'s time decay adjustment operation. This guarantees each track's relevance is only attenuated from the last point the user showed interest in the song by streaming it. The stream count factor determines the percentage relevance gained by a track for each average stream per day elapsed since track s was last listened to or analysed by TASTES, depending on the fraction returned by $days\_elapsed_i(s)$, while the decay factor defines the attenuation

of relevance per same day elapsed. The final values for ϕ_{decay} and η_{stream} were respectively chosen under the assumptions that it should take about twenty days for a track which is not listened to by the user to become virtually irrelevant, and about two streams per day elapsed to become twice as relevant. They are included in Table B.1. Our preliminary assessment of TASTES functionalities confirmed these values produced acceptable results for the testers. Additionally, they revealed that using a .05 lower ϕ_{decay} for tracks $s \in P_{n-1} \setminus P1_{n-1}$ was more appropriate, given the user's interest for tracks not in the playlist dwindles more steeply. Finally, tracks whose relevance values have dipped under 10% of the average relevance in the model set are brought back up to the value of .1 if they were streamed since the previous TES. This way, TASTES ensures the user only has to listen to a track once for it to become somewhat relevant again.

Seed Success Adjustment

Every track suggestion generated by TASTES is originally collected through a SWAPI recommendation request based on a playlist model track seed, as we detail ahead in the **Recommendation Generation** process. To account for seed success, all relevance ratings of seeds which originated the previous recommendation set are readjusted as follows:

$$U_{i,seed} = U_{i,seed} \times (1 + \zeta_{ss} \times success(seed)) \tag{3.7}$$

where the seed success factor ζ_{ss} determines the impact on relevance caused by the seed's success score $success(seed)$, which is determined by:

$$success(s) = \frac{\theta_{add} \times n_{add}(s) - \theta_{rej} \times n_{rej}(s) + \theta_{like} \times n_{like}(s) + \theta_{str} \times n_{str}(s)}{n_{total}(s)} \tag{3.8}$$

where $n_{add}(s)$, $n_{rej}(s)$, $n_{like}(s)$ and $n_{str}(s)$ are integers expressing the number of recommendations produced through seed s which were added to the user's playlist, rejected from the radio playlist, liked and streamed by the user, respectively, and $n_{total}(s)$ gives the total number of previous suggestions obtained using the seed. The categories are over-imposing in the order they are listed, meaning a track added, for instance, does not count as rejected, liked or streamed even if the user has also attributed those reactions. The same goes for the rest of the interaction hierarchy. The track weight factors θ_{add}, θ_{rej}, θ_{like}, and θ_{str} define the influence of each added, rejected, liked, and streamed recommendation produced by a seed in determining its success score.

The values of θ_{add}, θ_{like} and θ_{str} were only ascertained at the end of the preliminary assessment, during which this and the following operation were left out of the relevance update process. The values were determined based on the average normalised relevance of liked, added and streamed songs, respectively, in the preliminary testers' model track sets at the end of the initial evaluation period, and

later adjusted according to their feedback. As for the rejected weight factor θ_{rej}, its value was initially assigned based on the assumption that producing two rejected recommendations should nullify the success of producing one added recommendation, thus $\theta_{rej} = -\frac{\theta_{add}}{2}$. Nevertheless, we had to slightly adjust this value after realising the number of rejected tracks produced by the preliminary testers was often three times greater than the number of added recommendations. Finally, the seed success factor ζ_{ss} was determined based on the premise that each seed's relevance should increase by half per two recommendations added as a result of using the seed, or any equivalent outcome, given one addition may not be enough to confirm the track's aptitude as a seed. The values used for each factor in our final implementation of the strategy are included in Table B.1. A seed s is said to have been successful when $success(s) > 0$ and unsuccessful when $success(s) < 0$.

Relevance Refactorisation

Once time elapsed and seed success have both been accounted for, TASTES normalises and refactors all the relevance ratings $U_{i,s}$ so that each type of track in the model maintains its relative relevance reflected by the totality of the user's most recent feedback. The values are set as follows, from highest to lowest relevance:

1. $U_{i,s} \geq \theta_{added}$: Newly added tracks (A_i), from highest to lowest $SC_{i,s}$.

2. $\theta_{added} \geq U_{i,s} \geq \theta_{success}$: Successful seeds, from highest to lowest $U_{i,s}$.

3. $\theta_{success} \geq U_{i,s} \geq \theta_{liked}$: Liked recommendation (LR_i), from highest to lowest $SC_{i,s}$.

4. $\theta_{liked} \geq U_{i,s} \geq \theta_{model}$: All model tracks, excluding successful and unsuccessful seeds, from highest to lowest $U_{i,s}$.

5. $\theta_{model} \geq U_{i,s}$: Unsuccessful seeds, from highest to lowest $U_{i,s}$.

The strategy does not differentiate between the sets of added tracks originating from TASTES' recommendations ($s \in AR_i$) or from other sources ($A_i \setminus AR_i$), given there is no way of knowing the user's relative affinity toward each group. The relevance thresholds defined by the track weight factors θ_{added}, $\theta_{success}$, θ_{liked} and θ_{model} guarantee each track category's relative relevance is maintained while still supporting differentiation within the groups, thus ensuring the values are reflective of the user's most recent feedback while still retaining relevant cumulative data. Similarly to θ_{added} and θ_{liked}, the values used for $\theta_{success}$, θ_{model} were determined at the end of our preliminary evaluation by analysing the average relevance of successful seeds and model tracks respectively. They are presented in Table B.1.

3.2.3.E Recent Trend Extraction

Once all model track relevances U_i have been adjusted, TASTES' following task is to ascertain the user's recent relative facet inclinations within the now updated facet model P_i. The strategy relies solely on newly extracted data for this calculation, given it should best reflect the recent state of the user's preferences. Thus, we define the sets of relevant model tracks and recent trend tracks as $MT_i = \{s \in \{P_{i-1} \setminus X_i\} : SC_{i,s} > 0\}$ and $RT_i = \{MT_i \cup A_i \cup LR_i \cup RR_i \cup X_i\}$, respectively. The user's recent trend distribution vector t_i, where each entry $t_{i,f}$ is a value between 0-1 representing the user's recent affinity towards facet $f \in F$ at the time of TES_i, is determined as follows:

$$t_{i,f} = \sum_{s \in RT_i} \left(\frac{KNN_{i,f}(s)}{k} \times W_{i,s} \times \phi_{decay}{}^{days\_elapsed_i(s)}\right), \quad \forall f \in F \tag{3.9}$$

where $KNN_{i,f}(s)$ gives the number of k-nearest-neighbors of track $s \in RT_i$ in model track set P_i belonging to facet $f \in F$. The value used for k is the same as in the Similarity Threshold Calculation. The influence weight $W_{i,s}$ expresses how representative track s is of the user's recent trends at the time of TES_i. A track's influence weight is defined by the type of feedback the listener has produced on the song, and therefore determined as follows:

$$W_{i,s} = \begin{cases} \theta_{added} + (1 - \theta_{added}) \times \frac{SC_{i,s}}{\max\left(\sum_{r \in A_i} SC_{i,r}, 1\right)} & \text{if } s \in A_i, \\ \theta_{liked} + (\theta_{added} - \theta_{liked}) \times \frac{SC_{i,s}}{\max\left(\sum_{r \in LR_i} SC_{i,r}, 1\right)} & \text{if } s \in LR_i, \\ \theta_{model} + (\theta_{liked} - \theta_{model}) \times \frac{SC_{i,s}}{\max\left(\sum_{r \in RP_i} SC_{i,r}, 1\right)} & \text{if } s \in MT_i, \\ \theta_{rem} & \text{if } s \in X_i, \\ \theta_{rej} & \text{otherwise.} \end{cases} \tag{3.10}$$

where $max(a, b)$ returns the maximum value between a and b, to avoid division by zero. This approach to recent trend extraction is similar to the one employed in Zhao et. al. [31]'s sequential recommendation approach, with the addition of a weighted interaction matrix $W_{i,s}$ defining the impact of different user-item interactions in the final estimation and a time decay attenuation factor, determined similarly as in the relevance time decay adjustment process. TASTES does not differentiate between tracks added to the user's playlists originating from its recommendations and ones deriving from other sources, as there is no reason to assume that one group is more representative of the user's interests than the other. Similar to the methodology used in the relevance refactoring process, the weight factors θ_{added}, θ_{liked}, and θ_{model} are also employed at this stage to ensure each type of track contributing to the calculation of the user's recent trend distribution $t_{i,f}$ maintains its relative influence over the final estimation. Hence,

a recommendation the user has added to their playlist, for instance, will always weigh more in the estimation than one they have only attributed a "like" to. The values for θ_{rem} and θ_{rej} are both negative with $\theta_{rem} < \theta_{rej}$ given that a track removed from the user's playlist is more representative of what they are not interested in listening than a rejected recommendation. Accordingly, the value of θ_{rem} was initially set to double the value of θ_{rej}, which seemed to produce good results in the preliminary assessment. The final values used for all the aforementioned weight factors are included in Table B.1.

3.2.3.F Facet Preference Distribution Estimation

The following procedure in the TES processing pipeline is to estimate the listener's facet preference distribution vector p_i using the information retained on their past trends p_{i-1} and the just determined recent preference distribution vector t_i. This is a simplified version of the model adjustment process employed in Zhao et. al.'s interest-aware solution [31]. The user's facet preference distribution is updated as follows:

$$p_{i,f} = \begin{cases} t_{i,f}, & \text{if } days\_elapsed_i \geq 20, \\ & \forall f \in F \\ (1 - \gamma \times days\_elapsed_i) \times p_{i-1,f} + \gamma \times days\_elapsed_i \times t_{i,f}, & \text{otherwise.} \end{cases} \tag{3.11}$$

where γ represents the facet preference distribution daily update factor, and $days\_elapsed_i$ gives the fraction of days passed since the last iteration step TES_i. The more days have passed since TASTES' previous analysis was performed, the less relevant the strategy's retained preference distribution p_i becomes relative to the recent trend distribution vector t_i in determining the user's updated facet preferences. The value of γ was set according to the same methodology employed when determining the relevance decay factor ϕ_{decay} of a track, in the relevance time decay adjustment process: assuming the user's past preferences should become irrelevant if $days\_elapsed_i \geq 20$ then the daily update factor should be $\gamma = .05$, as presented in Table B.1.

3.2.3.G Recommendation Generation

With the totality of the user preference model structure updated, TASTES proceeds to the final stage of its processing pipeline, where a new set of recommendations is produced based on the adjusted data. The approach taken at this stage was heavily influenced by the recommendation selection methodology employed in Zhao et. al.'s *TecRec* strategy [31] which relies on a baseline recommender system to produce an initial set of rated suggestions before re-ranking them according to an objective function which balances the desired accuracy and diversity of the final set.

Initial Pool Collection

The strategy's first step in generating new suggestions is to retrieve the initial set of tracks for recommendation I_i through SWAPI recommendation requests, as defined in Section 2.1.2.C. TASTES iteratively constructs set I_i by performing successive requests of size N_REQ. Each request is performed using a single and distinct model track $s \in P_i$ as seed, based on their current relevance rating $U_{i,s}$. A new seed is selected before every request according to a weighted probability distribution, calculated as follows:

$$prob\_seed_i(s) = \frac{U_{i,s}}{\sum_{c \in P_i} U_{i,c}}, \quad \forall s \in P_i \tag{3.12}$$

Each request is made using only one seed so that every suggestion can be traced back to one model track, allowing the strategy to properly assess its success, in the following iteration, and adjust its relevance rating according to the results produced. After every new group of N_REQ recommendations is retrieved, all tracks whose IDs are included in the taboo list TB_i are discarded, before adding the remaining to I_i. The strategy will continue performing requests until it has collected a total of $|I_i| = N\_REQ \times N\_FINAL$ tracks, where $N\_FINAL = |R_i|$ is the intended size of the final set of recommendations R_i. The reason for this set limit is twofold: SWAPI's restrictions on the rate of requests and number of retrieved tracks permitted in a short time frame, and the increase in computational costs imposed by the subsequent operations when using a large initial candidate pool. The values used for N_REQ and N_FINAL in our final evaluation are included in Table B.1.

Baseline Rating

Before selecting the final recommendation set R_i the strategy first computes a baseline rating vector B_i where each entry $B_{i,r}$ is a value between 0-1 expressing how fitting each track in the initial recommendation pool $r \in I_i$ is for the continuation of the user's playlist. A baseline rating of 0 and 1 are respectively attributed to the tracks in the pool which are least and most likely to fit in the musical area of interests encompassed by the user's playlist. Accordingly, a track's baseline rating is determined by its closeness to the songs in P_i in the audio feature space under the user's perception of similarity, as follows:

$$B_{i,r} = \sum_{s \in P_i} kmd(r,s) \times U_{i,s}, \quad \forall r \in I_i \tag{3.13}$$

where the k-membership distance function, kmd, is defined by:

$$kmd(r,s) = \begin{cases} \frac{D_{i,s} - sim_i(r,s)}{D_{i,s}} & \text{if } sim_i(r,s) < D_{i,s}, \\ 0 & otherwise. \end{cases} \tag{3.14}$$

where $sim_i(s, r)$ represents the weighted distance between recommendation r and model track s in the audio feature space, given by Equation (3.2). The baseline rating determination process was defined based on TASTES's KNN framework for similarity: a recommendation track $r \in I_i$ is similar to a model track $s \in P_i$ if $r \in P_i \implies r \in KNN_i(s)$, where $KNN_i(s)$ is the set of track s's k most similar tracks in the model set and k assumes the same value used in the Similarity Threshold Calculation. Additionally, the closer recommendation r is to a similar playlist model song s, and the higher the relevance rating of that song $U_{i,s}$ is, the more potential track r is of being a good candidate for the playlist's continuation, expressed accordingly by a higher baseline rating $B_{i,r}$.

Post-Ranking Selection

The selection of the final set of recommendations $R_i \subset I_i$ is centred on balancing the two main objectives of the proposed strategy: catering to the user's most updated trends within the area of interest associated with the playlist in focus, for the purposes of producing relevant songs for PC, and selecting music that pushes the listener into expanding the area's boundaries. To accomplish this, TASTES constructs the final suggestion set R_i by iteratively selecting the best candidate $r \in I_i$ based on its value of inclusion $V_{i,r}$ to the final set. Following the approach's goals, a candidate r's inclusion value is determined by the tradeoff between its aptitude for continuation, measured through its baseline rating $B_{i,r}$, and its potential for interest expansion, measured through the incompatibility between its k-neighborhood facet distribution in P_i, represented by vector $N_{i,r}$, and the user's updated facet preference distribution, expressed by vector p_i. Assuming TASTES intends to produce a final set of recommendation R_i of size N_FINAL, the re-ranking process is as follows:

1. Start with an empty final recommendation set $R_i = \{\}$.

2. Determine the value of inclusion $V_{i,r}$ for every track in I_i according to the following objective function:

$$V_{i,r} = \lambda \times \sum_{s \in R_i \cup \{r\}} \frac{D_{i,s}}{|R_i| + 1} + (1 - \lambda) \times kld(p_i \| \sum_{s \in R_i \cup \{r\}} \frac{N_{i,s}}{|R_i| + 1}), \quad \forall r \in I_i \tag{3.15}$$

   where $kld(a\|b)$ gives the Kullback-Leibler Divergence (KLD), introduced in the previous chapter, between distributions a and b and $N_{i,s}$ is the k-neighbourhood facet distribution of track s in model P_i, given by:

$$N_{i,s} = \frac{1}{k} \begin{pmatrix} KNN_{i,f_1}(s) \\ KNN_{i,f_2}(s) \\ \cdots \\ KNN_{i,f_{|F|}}(s) \end{pmatrix} \tag{3.16}$$

   $KNN_{i,f}(s)$ gives the number of k most similar tracks to s in set P_i belonging to playlist facet $f \in F$, with k being the same value used in the remaining KNN operations performed by the

strategy. This calculation is analogous to the one represented in Equation (2.1), employed in Zhao et. al.'s solution [31] and in line with the multi-labelled classification framework proposed by Scaringella et. al. [13]. The continuation/expansion tradeoff factor λ balances the relative weight of the two aspects which determine the inclusion value: the average baseline rating of R_i after adding a candidate r to the set, and the KLD between the user's preference distribution p_i and the average facet distribution $N_{i,s}$ of tracks in set $R_i \cup \{r\}$. Both aspects are always determined and normalised to a 0-1 scale for every track $r \in I_i$, at every post-ranking step, before calculating all inclusion values.

3. Add recommendation $r \in I_i$ with the highest value of inclusion $V_{i,r}$ to the final set R_i, removing it from the initial pool set I_i.

4. If $|R_i| < N\_FINAL$ then return to step 2, else output R_i as the final recommendation set.

Equation (3.15) displays a similar structure to the objective function employed in TecRec's re-ranking process [31]. Nevertheless, given TASTES is an expansion strategy, it strives for final set facet distributions which escape the user's relative preferences instead of attempting to approximate them like Zhao et. al.'s solution. Consequently, our approach attempts to maximise the kld between the distributions instead of minimising it. The continuation/expansion tradeoff factor λ in Equation (3.15) determines the importance attributed by TASTES to maximising the total relevance of R_i relative to the set diverging from the user's estimated preference distribution p_i. The lower λ is, the further the final set's facet distribution will be to p_i, benefiting exploration to the detriment of a less relevant set of final recommendations. A high λ value will on the other hand produce a set of more relevant tracks for the continuation of the user's playlist, with a facet distribution which is closer to the user's relative affinities, expressed by p_i. Given the restricted time window available for our preliminary evaluation we were unable to undergo further experimentation with different values of λ. Thus, the parameter was set to .50, which is the same value used in our final implementation, as it seemed to have produced good results in the preliminary assessment.

Once R_i is finally ascertained, TASTES will update the user's radio playlist, included in their *Spotify Library*, with the newly generated recommendations using the appropriate SWAPI requests. The set's data is retained in TUDB, including the seeded model tracks used to produce each suggestion selected, for the purpose of seed success analysis in the following TES. Lastly, all track IDs in R_i are added to the taboo set TB_i, concluding the iteration step processing pipeline.

3.3 SRSE

Spotify Recommendation Strategy Emulator (SRSE) is a sequential recommendation approach which operates similarly to TASTES but without employing its interest-aware and taxonomy-free framework for preference modelling. Instead, the strategy's objective is simply to produce the most relevant suggestions at each iteration according to *Spotify*'s methodology, employed through the available SWAPI services. SRSE and TASTES operate separately and are never used simultaneously for the continuation of the same playlist. In this section, we detail the strategy's architecture, based on how it differentiates from the proposed solution at each phase of its operational timeline.

3.3.1 Retained Data Structures

SRSE only requires three of the data structures employed by its counterpart to produce PC suggestions:

- The **model track set** vector P_i containing all the *Spotify* audio feature data for each relevant track in the area of interest bound to the user's playlist, free of any corresponding facet classifications.
- The **relevance ratings** vector U_i, expressing the user's affinity towards each track $s \in P_i$.
- The **tabu set** vector TB_i, to avoid any repetition in the recommendation sets produced.

The data structures listed follow the respective architectural description provided in 3.2.1. All data is maintained through TUDB, in a similar fashion to TASTES.

3.3.2 Setup Step

Similarly to TASTES' approach, SRSE's initial execution step is responsible for initialising the data structures maintained by the recommendation strategy to model the user's preferences. The strategy's setup procedure is virtually identical to the interest-aware approach up until, and including, the Relevance Ratings Update stage. After that, the strategy moves on straight to the SRSE Recommendation Generation stage described ahead.

3.3.3 Standard Processing Pipeline

In the iterations that follow the initial setup step, SRSE implements a simplified version of its competing strategy's processing timeline. The aim of this design is only one: employing the recommendation functionalities offered through SWAPI to emulate *Spotify*'s methodology for producing relevant recommendations in the same operational environment as TASTES. Accordingly, each execution step performed using SRSE follows the subsequent structure:

1. Feedback Extraction.
2. Relevance Adjustment.
3. Recommendation Generation.

3.3.3.A Feedback Extraction

SRSE begins each processing cycle by extracting all the new data generated by the user since the previous iteration was completed, similar to our taxonomy-free solution. This stage of the process is identical to the procedure described in Feedback Extraction. The same relevant data is collected pertaining to modifications to the user's playlists, interactive responses to individual recommendations produced in the previous execution step and recent *Spotify* streaming activity. Sets P_i and TB_i are then adequately updated with the newly collected data.

3.3.3.B Relevance Adjustment

Once the retained preference model set has been appropriately updated, SRSE moves on to adjust the relevance ratings of all tracks in P_i. The first stage of this process is identical to TASTES' time decay adjustment procedure, described in 3.2.3.D, where the ratings vector U_i is updated according to the time elapsed since the last iteration of the strategy and the recently streamed vector $SC_{i,s}$ produced in the previous stage of the pipeline. After time decay has been accounted for, SRSE normalises and refactors all relevance $U_{i,s}, \forall s \in P_i$, in the same fashion as its counterpart, but employing only three value tiers instead of five, from highest to lowest relevance:

1. $U_{i,s} \geq \theta_{added}$: Newly added recommendations, from highest to lowest $SC_{i,s}$.
2. $\theta_{added} \geq U_{i,s} \geq \theta_{liked}$: Liked recommendations, from highest to lowest $U_{i,s}$.
3. $\theta_{liked} \geq U_{i,s}$: All remaining model tracks, from highest to lowest $U_{i,s}$.

The values used for the track weight factors θ_{added} and θ_{liked} were the same for TASTES' and SRSE's final implementations. Once the relevance ratings vector has been updated, the strategy moves on to the recommendation generation stage.

3.3.3.C Recommendation Generation

After adjusting the relevance ratings U_i, the strategy is ready to produce the final set of recommendations R_i. Just like TASTES, SRSE employs single seeded SWAPI recommendation requests to produce the suggestion, with the seed selection function being the same as the one described by equation 3.12. However, the strategy differs from its interest-aware counterpart by soliciting a single suggestion per

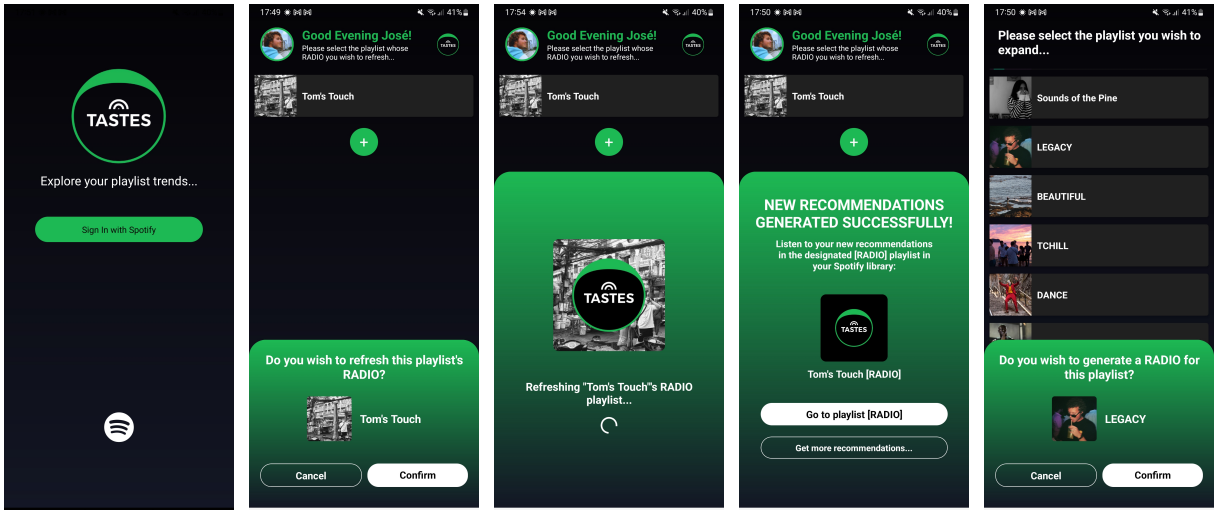
request, i.e. $N\_REQ = 1$. The purpose of this approach is to ensure that only the most relevant track is generated by *Spotify* for every request made, so that R_i is contrived of *Spotify*'s top recommendation picks for each of the distinct seeds used. SRSE will perform these requests until it has obtained a total of N_FINAL tracks, where N_FINAL represents the size of the final set R_i the strategy intends to produce. After each request is performed, if the track ID of the suggestion returned by SWAPI is already included in the tabu set TB_i, that track is discarded. Once R_i is fully obtained, the user's radio playlist is updated with the newly produced recommendations. All the IDs of the recommendations in R_i are added to the tabu set TB_i, so that they are not repeated in future recommendation sets, concluding SRSE's processing cycle.

3.4 Technology

For the purposes of conducting the comparative assessment which informed the evaluation of the research questions posed in this project, two *Python* applications were developed, employing each of the recommendation methodologies introduced. The version of *Python* used was 3.11.6 and the primary Integrated Development Environment (IDE) chosen for development was *Microsoft*'s *Visual Studio Code*. The implementations employ the following *Python* external libraries:

- *Flask* (v3.0.2): Used to build and deploy the Python server application.
- *spotipy* (v1.23.0): Used for SWAPI endpoint communication.
- *SQLAlchemy* (v2.0.27): Used to communicate with TUDB.
- *numpy* (v1.26.4): Used for data structure formatting and handling.
- *sklearn* (v1.4.1.post1): Used for K-Means classification and all KNN calculations.
- *tensorflow* (v2.15.0): Used to calculate KLD between distribution vectors.

TUDB was implemented in *PostgresSQL* and maintained using the *ElephantSQL* management system. As an initial approach to development, a *JavaScript* frontend was designed, using the *Expo* framework, to trigger each strategy's execution steps and input the necessary *Spotify* user and playlist data through a mobile interface. Given the program proved to be too volatile to be distributed to participants of our evaluation study, detailed in the following chapter, it was instead used to manually trigger each TASTES and SRSE iteration steps, remotely, during the course of the study. This decision was also supported by the limited time and resources available for the practical assessment and the need to control the strategies' rate of execution during the evaluation process.



(a) Login Screen (b) User Playlist Screen (c) Refresh Screen (d) Radio Playlist Screen (e) Add Playlist Screen

Figure 3.3: TASTES Mobile Application Screenshots

4

Evaluation

Contents

The following chapter describes the procedure taken to evaluate the proposed solution. We begin by describing the practical approach taken to assess the research questions posed in this project, using the recommendation frameworks detailed in the previous chapter. We also describe the experimental setup devised for our comparative analysis of the TASTES and SRSE's aptitude for PC on *Spotify*. Secondly, we detail the preliminary testing process which informed the necessary adjustments made to our implementation and methodology, leading up to the primary testing phase. Finally, we provide a comprehensive description of the structure of the user study, conducted as the primary means of evaluation, as well as the data collected to instruct our subsequent analysis.

4.1 Approach

To assess TASTES's capabilities for recommendation and taste expansion, we devised an evaluation framework centred on capturing the strategy's performance for PC on *Spotify*. The decision to tackle this particular task stemmed from the possibility of addressing both the initially posed research questions simultaneously:

1. How effective is a taxonomy-free interest-aware recommendation approach such as TASTES at expanding the listener's musical interests?

2. Can a taste expansion approach to recommendation improve listeners' music discovery experience?

Firstly, we aimed to gauge the strategy's ability to generate relevant suggestions beyond the confines of the musical themes represented in a playlist. We sought to understand the impact of these picks on the diversity and coverage of the tracklist, as well as on the listener's tendencies and perceptions towards their playlist. This informed our analysis in regard to effectiveness in broadening musical interests. Additionally, as music discovery is intrinsic to PC tasks, we aimed to study the effects produced by TASTES on the listener's overall satisfaction with the resulting recommendation experience. This second aspect allowed us to assess its competence in facilitating music discovery.

To establish a robust baseline for the evaluation of our taxonomy-free approach, we designed another sequential recommendation tool employing *Spotify*'s recommendation functionalities, available through SWAPI, for PC. The SRSE, introduced in Section 3.3, acted as our solution's counterpart in the experimental assessment, setting the standard upon which we measured TASTES' performance. Ultimately, by juxtaposing the results produced by both methodologies in the same experimental settings, we aimed to determine the potential of the taxonomy-free approach for enhancing music discovery tasks on *Spotify* and pushing the user towards new areas of interest.

4.1.1 Experimental Setup

In accordance with the approach taken for evaluation, the testing setup was designed specifically to emulate a PC scenario on *Spotify*. The primary concern was to devise a workflow that allowed testers, all of whom would be *Spotify* users, to access and react to recommendations through their existing profiles, on the application, without the need to interact with any other external tool. Furthermore, TASTES and SRSE are sequential online recommendation strategies. This means they employ an iterative framework to process feedback and re-adapt to the user's evolving preferences whenever a new processing cycle is triggered. As mentioned in Section 3.1, this design was chosen to maximise the limited functionality offered through SWAPI. The experimental setup was built to accommodate this particular methodology.

Accordingly, the testing process was contrived of cycles, referred to as Recommendation-Feedback Cycles (RFCs), which followed the subsequent structure:

1. A set of recommendations is produced by the employed strategy, TASTES or SRSE, by leveraging the user's most recent preference data.
2. The recommendations are published in the user's *Spotify Library*.
3. The user provides direct, individual feedback in response to the song suggestions, through interactions supported by *Spotify*'s application interface.
4. When the user is ready to receive more recommendations, a new processing cycle of the employed strategy is manually triggered by the evaluating party.
5. The new direct and indirect feedback is processed through the employed approach, updating the user's preference model, and producing a new set of song recommendations.

The complete evaluation procedure involved putting *Spotify* users through a series of RFCs while one of the tested methodologies is employed, then undergoing the same process using the second approach. By collecting the same relevant data for each of these segments we could measure TASTES and SRSE's relative performance, thereby informing our initially posed research questions.

4.1.2 Personal Playlist

As a requirement for PC tasks, a list of tracks must be selected as the target for which the employed strategy attempts to tailor the generated selections. Hence, before the start of the evaluation process, a playlist curated by the tester and included in their *Spotify Library* is chosen to be the focus of the test. This playlist remains the same for each test participant throughout the entire evaluation period so that both approaches are tested in the same conditions. We refer to it as the participant's **personal playlist**. They define the area of musical interests being catered to, and consequently, the same area our TASTES aims to expand.

Besides being created by the user, for the purpose of isolating a self-defined plane of their preferences, the only other requirement for the personal playlist is a minimum length of 20 distinct tracks. The objective of this restriction is to guarantee enough data is available for TASTES to build its initial facet model, as detailed in Section 3.2.2.D.

4.2 Preliminary Testing

In preparation for the primary evaluation, a short preliminary assessment was conducted with three major objectives:

- Calibrate the control parameters of the developed methodologies listed in Table B.1, to maximise their performance in the limited time frame available for primary assessment.
- Test all components of TASTES' and SRSE's architecture practically and confirm the entire processing pipeline works as intended.
- Gauge the perceived impact of employing the taxonomy-free approach to recommendations, as well as its potential for interest expansion.

We selected 3 candidates for initial testing, 1 female and 2 male, between the ages of 23-25. The testers were chosen based on availability, usage frequency, and familiarity with the *Spotify* application, to maximise feedback data production in a short interval. We asked each tester to select a personal playlist from their *Spotify Library*, entirely curated by themselves, which they listen to often. The prevailing genres of the selected playlists were Hip-Hop (313 tracks long), Electronic (156 tracks long), and Indie Pop (72 tracks long).

4.2.1 Testing Process

The preliminary tests were comprised of 6 successive RFCs, divided into 2 series of 3. During each segment, testers were exposed to 3 sets of 20 track recommendations produced using either TASTES or SRSE. The order of the tested strategies was randomly chosen and unbeknownst to the candidates. At the end of each RFC, the suggested songs were published directly to the user's *Spotify Library*, through SWAPI, in the form of a playlist solely dedicated to this purpose. We refer to it as the user's **radio playlist**. Figure 4.1 illustrates an example of the user's view when accessing the application during testing, with the personal ("Tom's Touch") and radio playlist ("Tom's Touch [RADIO]") appropriately highlighted.

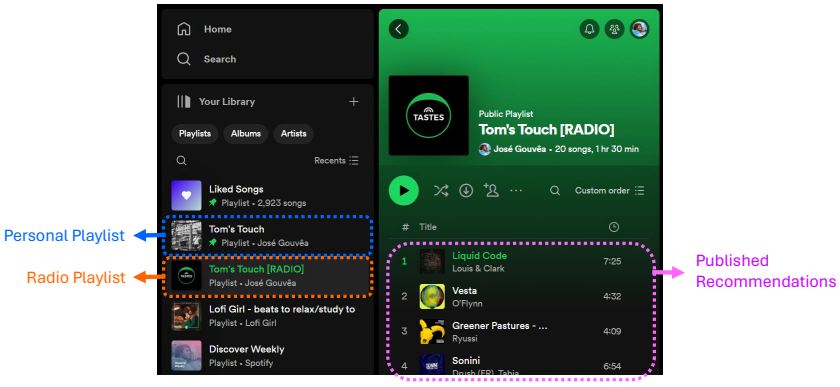


Figure 4.1: *Spotify Library* User View Example

Before the start of testing, the applicants were also informed regarding the 4 possible feedback responses available for recommendations:

- Attributing a ”like” to a track, adding it in the user's *Spotify Library*, would signify the tester is interested in the recommended song but not necessarily for the purpose of continuing their personal playlist.
- Adding the track to the personal playlist would indicate the user is interested in the song for the purpose of their playlist's continuation.
- Removing the track from the radio playlist would indicate the user is not interested in that recommendation.
- Not attributing any of the above feedback responses would mean the user is not completely uninterested in the suggestion but is also not interested enough to warrant a positive response.

Concerning the frequency of RFCs, we asked the participants to notify us of when they had finished reacting to a set of recommendations and wished to receive a new one. This was done with the intention of determining the optimal time required for users to produce the necessary feedback during each RFC, thus informing the appropriate interval to choose for primary evaluation. Upon receiving confirmation from a participant, a new set of 20 recommendations was generated using whichever strategy being tested in that segment. The corresponding radio playlist in the user's *Spotify Library* is then updated with the new song picks, triggering the start of a new RFC.

The total duration of the preemptive assessment was 8 days. Given the nature of this initial testing phase, no formal data analysis was performed on the strategies' relative ability for music discovery. The focus was on guaranteeing the best conditions for the primary user study and confirming the functionality of the developed architectural components. Some feedback from the 3 preliminary testers was collected informally, all of which confirmed a noticeable difference between the suggestion sets produced through TASTES and SRSE. The feedback was also taken into consideration in determining the necessary adjustments to our testing methodology.

4.2.2 Resulting Adjustments

The main goal at this stage was to adjust TASTES' and SRSE's architectural parameters to maximise the comparative data produced in the primary evaluation study. In B.1, we display the final values used for each factor during the study, informed by the results of the initial tests.

In regard to the defined assessment framework, some design corrections were also considered. Namely, we opted to reduce the size of the recommendation sets produced per RFC to 15 and settled on a 3-day time frame for the users to listen and react to each set. Additionally, we developed a comprehensive guideline document to act as a reference for participants during the main study. The document is presented in Appendix B.

4.3 User Study

As the primary method for evaluating our approach, we conducted a 30-day within-subjects study with a total of 20 participants. The aim of the study was to expose each tester to two identical series of RFCs while employing each recommendation strategy, similarly to the preliminary testing approach, and collect the results produced in each set for our final analysis.

4.3.1 Sampling

Over 30 candidates were interviewed for the evaluation study, all *Spotify* users with *Premium* subscriptions and at least one manually curated playlist in their *Library*. The candidates were pruned according to the following requirements:

- Minimum of one year of active usage as a *Spotify* subscriber.
- Minimum of 3 hours listening to the personal playlist per week.
- Minimum of 20 tracks in the personal playlist.
- Availability for the entire period of the study.

A total of 20 candidates were selected for the final study sample.

In regards to personal playlist selection, the candidates were asked to pick playlists they listened to as regularly as possible and preferably during the same activity. TASTES and SRSE both leverage recent *Spotify* streaming data to track the user's most recent inclinations. Hence, we aimed to maximise the data produced by each participant during the study by guaranteeing they interact with their playlists frequently. Furthermore, ensuring each participant mostly listens to their personal playlist while undergoing the same activity allows for further control of contextual variables that may influence the type of recommendations the user is receptive to. This way, the focal area of the user's interests, which the strategies attempt to cater to, remains the same throughout the entire study, contributing to a fairer assessment of the strategies' performance in the same experimental context.

4.3.2 Procedure

The approach taken to the primary evaluation was analogous to the one used in preliminary testing. The study was conducted over a 30-day stretch, divided into two consecutive 15-day segments. During each of the segments, every participant was exposed to 5 sets of 15 recommendations produced using one of two tested strategies every 3 days, following the conclusions taken from the initial assessment. This meant every segment was composed of five 3-day-long RFCs. In order to control for variables resulting from users testing one strategy before or after the other, the participant group was also split in half.

During the first 15-day segment of the study, half of the participants were exposed to track suggestions produced through SRSE, followed by a transition to recommendations generated using the TASTES in the latter half of the study. Conversely, the second subgroup of participants experienced the reverse sequence.

Before the start of the study, all applicants were handed a document with the necessary guidelines for testing, presented in Appendix B. The document includes the following elements:

- Short introduction to the TASTES recommendation strategy.
- Overview of the study's structure and timeline.
- Explanation of the direct feedback process using *Spotify* application interface interactions.
- Some important usage considerations.

Participants were told they would be testing two distinct configurations of TASTES for each 15-day segment of the study. For the purposes of simplicity and fairness, both tested strategies were introduced as two versions of the same recommendation approach, referred to in the document as "TASTES-A" and "TASTES-B". No comparative description of the configurations was provided either, to avoid any possible bias that could jeopardise the legitimacy of the evaluation results. This conceptual framework was maintained throughout the entirety of the study so that all participants remained impartial in their comparative assessments.

4.3.3 Data Collection

During the study, the same three types of data were collected in each segment, to inform our relative assessment of the tested strategies:

- **Playlist Audio Composition Data**: Collected at the start and end of each segment, to track changes to the playlist's musical area of coverage resulting from the added recommendations.
- **Recommendation Interaction Data**: Collected at the end of each RFC, to measure the accuracy of the generated track suggestions.
- **User Feedback Data**: Collected through surveys filled out by the study participants at the end of each segment, to assess their satisfaction with the music discovery journey provided.

Subsequently, we describe the specific measurements taken for each data type, as well as the corresponding collection methodology employed.

4.3.3.A Playlist Data

We looked at two key aspects to assess the evaluated approaches' impact on the area of musical interest represented in the personal playlists. Firstly, we calculated the tracklist's audio feature variances before and after each study segment. The purpose of these measurements was to determine the relative variance changes caused by each tested strategy, to inform our interest expansion analysis from a quantitative perspective. Similarly, we employed TASTES' unsupervised classification method, described in Section 3.2.2.D, to determine each playlist's facet count before and after the segments, aiming to capture any changes in the facet count produced by TASTES and SRSE.

All playlist data was ascertained and catalogued at the end of the study, using the initial and final states of each playlist's track model, as defined in Section 3.2.1, available through the TUDB. The variances were calculated for all *Spotify* audio features listed in Table B.1.

4.3.3.B Recommendation Data

In regards to individual recommendation feedback responses, attributed by the participants through the *Spotify* application, the following data was retrieved during each segment of the study:

- N_ADDED_TOTAL_S / N_ADDED_TOTAL_T: Total amount of recommendations produced by SRSE / TASTES which the user added to their personal playlist.
- ADDED_FACTOR_S / ADDED_FACTOR_T: Percentage of songs added to the user's personal playlist which were recommendations produced by SRSE / TASTES.
- N_LIKED_TOTAL_S / N_LIKED_TOTAL_T: Total amount of recommendations produced by SRSE / TASTES which the user attributed a "like" to.
- N_REJ_TOTAL_S / N_REJ_TOTAL_T: Total amount of recommendations produced by SRSE / TASTES which the user removed from the radio playlist.

The data was logged into TUDB at the conclusion of each RFC. This process was purposefully integrated into both tested strategies' processing pipelines, as detailed in Section 3.2.3.G. Only the added factor was calculated at the end of the study, using the initial and final states of each candidate's personal playlist model to determine the percentage of track added originating from the user's radio playlist. These measurements served to quantitatively evaluate the SRSE's and TASTES' relative accuracy for continuation, by juxtaposing the results produced by both methodologies for the same user/playlist.

4.3.3.C User Feedback Data

In addition to assessing the approaches' performance from a quantitative standpoint, we also wished to compare their perceived impact on music discovery from the user's perspective. To do so, we con-

structed two surveys for participants to complete during the study. The first survey was administered at the outset of the evaluation to gather pertinent descriptive data regarding the participant sample's music consumption trends. The second survey was filled out by each tester twice, at the end of each segment of the study, with the aim of capturing their feedback pertaining to different aspects of the recommendation experiences undergone.

A – Initial Profiling Survey

Before the start of the study, participants filled out a preliminary questionnaire pertaining to their demographics, preferences as music listeners, *Spotify* usage habits, and the personal playlists selected for the study. The survey's objective was to catalogue relevant user profiling data that could provide insight into the prevailing characteristics of the study sample as music listeners and *Spotify* users. Furthermore, we intended to ascertain any possible correlations between these characteristics and the responses given in the subsequent satisfaction feedback surveys, allowing us to better understand the reasoning behind them. In Figure 4.2, we detail the list of questions in the initial survey, including the corresponding response forms employed.

TASTES Evaluation Study - Initial Survey Questions

Question ID	Question	Answer Type	Scale/Options
ID1	Gender	Multiple Choice	Female / Male / Other
ID2	Age	Short Answer	-
IM1	State your agreement with the following statement: "Music plays a significant role in my day-to-day life."	Likert Scale	1 (completely disagree) - 5 (completely agree)
IM2	State your agreement with the following statement: "My musical preferences align closely with those of the majority of music listeners."	Likert Scale	2 (completely disagree) - 5 (completely agree)
IM3	State your agreement with the following statement: "I always prefer to listen to music I have not heard before."	Likert Scale	3 (completely disagree) - 5 (completely agree)
IM4	State your agreement with the following statement: "When I find a new song I enjoy, I do not get tired of hearing it over and over again."	Likert Scale	4 (completely disagree) - 5 (completely agree)
IM5	How familiar would you say you are with the Spotify application and its functionalities?	Likert Scale	5 (completely disagree) - 5 (completely agree)
IS1	For how long have you been a Spotify user?	Short Answer	-
IS2	How familiar would you say you are with the Spotify application and its functionalities?	Likert Scale	1 (extremely unfamiliar) - 5 (extremely familiar)
IS3	On average, how many hours a week do you spend listening to music on Spotify?	Short Answer	-
IS4	How much time do you spend listening to playlists created by yourself?	Likert Scale	1 (none of the time / 0%) - 5 (all the time / 100%)
IS5	How often do you normally add new songs to your Spotify playlists?	Multiple Choice	Almost never / Every month / Every week / Every other day / Every day
IS6	How would you define your ideal playlist in terms of diversity of music styles or genres?	Likert Scale	1 (extremely homogeneous) - 5 (extremely diverse)
IS7	According to your own experience, rate Spotify's ability to offer song recommendations that fit your music taste.	Likert Scale	1 (rarely capable) - 5 (consistently capable)
IP1	Rate your playlist in terms of the diversity of music styles or genres in it.	Likert Scale	1 (extremely homogeneous) - 9 (extremely diverse)
IP2	State your agreement with the following statement: "Listening to this playlist can often feel repetitive."	Likert Scale	1 (completely disagree) - 9 (completely agree)
IP3	What percentage of this playlist was a result of Spotify's recommendation system rather than individual selection?	Likert Scale	1 (all individually selected / 0%) - 9 (all recommended / 100%)
IP4	Rate the playlist in terms of your own listening experience.	Likert Scale	1 (worst) - 9 (best)

Figure 4.2: Initial Survey Questions

For the purposes of preserving anonymity and data protection, each study participant was assigned a standardised candidate ID, used in all data collection and treatment to identify the participant. Furthermore, to ensure transparency and compliance with ethical guidelines, detailed information regarding

participant consent and terms was provided at the start of the questionnaire. The survey was done through *Google Forms* and is presented in full in Appendix C.

B – User Feedback Survey

To collect the necessary feedback data for ascertaining the impact of the tested strategies from the user's perspective, we designed a questionnaire aimed at capturing the participants' impressions on different aspects of their recommendation experiences. Instead of assessing recommendation diversity from a quantitative standpoint, as the approach taken by Kaya and Bridge when evaluating their interest-aware solution [26], we aimed to measure this aspect from the perception of the user. Consequently, we included inquiries to capture not only the accuracy of the tested systems but also the diversity perceived by the user in the recommendation sets produced. The feedback surveys were filled out by all candidates at the end of each 15-day segment of the study, after undergoing 5 RFCs supplied by either TASTES or SRSE. The posed questions were the same for both segments, allowing us to make a 1-to-1 comparative assessment of the approaches' influence over each evaluated aspect of the PC experience. The feedback survey questions are presented in Figure 4.3.

TASTES Evaluation Study - Feedback Survey Questions

Question ID	Question	Answer Type	Scale/Options
FP_4	Rate the level of diversity of your whole playlist by considering the variety of music styles and genres it contains at this point.	Likert Scale	1 (extremely homogeneous) - 9 (extremely diverse)
FP_5	State your agreement with the following statement: "Currently, listening to my playlist evokes a sense of monotony or repetition."	Likert Scale	1 (completely disagree) - 9 (completely agree)
FP_6	Rate the playlist in terms of your own listening experience.	Likert Scale	1 (worst) - 9 (best)
FR_1	State your agreement with the following statement: "I had no trouble using the TASTES radio playlist on Spotify as it was instructed for the purposes of this study."	Likert Scale	1 (completely disagree) - 5 (completely agree)
FR_2	State your agreement with the following statement: "TASTES' recommendations covered all the music styles and genres present in my playlist."	Likert Scale	1 (completely disagree) - 9 (completely agree)
FR_3	State your agreement with the following statement: "I found TASTES' recommendations, belonging to music styles and genres already present in my playlist, interesting."	Likert Scale	1 (completely disagree) - 9 (completely agree)
FR_4	State your agreement with the following statement: "TASTES recommended music belonging to styles and genres not present in my playlist."	Likert Scale	1 (completely disagree) - 9 (completely agree)
FR_5	State your agreement with the following statement: "I found TASTES' recommendations, belonging to music styles and genres not present in my playlist, interesting."	Likert Scale	1 (completely disagree) - 9 (completely agree)
FR_7	Rate the level of diversity of the radio recommendations produced by this version TASTES, considering the variety of music styles and genres represented.	Likert Scale	1 (completely disagree) - 9 (completely agree)
FR_8	Rate this version of TASTES in terms of your overall satisfaction with the recommended songs.	Likert Scale	1 (extremely unsatisfied) - 9 (extremely satisfied)

Figure 4.3: Feedback Survey Questions

As previously mentioned, for the sake of keeping the participants' assessment unbiased and simplifying the instructions given during the study, we referred to TASTES and SRSE as two configurations of TASTES. Consequently, even though survey questions FR_1 to FR_8 only mention TASTES and not SRSE, from the candidates' perspective, each feedback survey referred to the version of TASTES they had been testing during the preceding 15-day segment. Thus, every participant provided feedback for both tested strategies knowing solely that they were assessing distinct recommendation methodologies. In our subsequent analysis, we differentiate the answers pertaining to each recommendation approach by adding an "S" or a "T" at the end of the question ID, indicating results produced by SRSE and TASTES respectively. For instance, questions FR_8S and FR_8T refer to the overall satisfaction

perceived by participants while testing SRSE and TASTES testing segments, respectively.

Furthermore, each participant was provided with their answers to the first feedback survey before filling out the second. This allowed the candidates to have a more accurate comparative assessment regarding the different aspects of the experience felt during each segment. With the knowledge of their previous ratings, differences in responses to the same inquiries become more indicative of the users' favour between each strategy's performance. Hence, we can better inform our comparative analysis from the user's perspective. The feedback questionnaire was performed using *Google Forms* and is presented in its totality in Appendix C.

4.3.4 Concluding Remarks

The evaluation process described in this chapter was designed to rigorously assess the capabilities of the TASTES recommendation strategy in expanding user musical interests and enhancing music discovery on *Spotify*. By comparing TASTES to a baseline approach employing *Spotify*'s recommendation methodology, SRSE, we aimed to address our two primary research questions regarding the effectiveness and user satisfaction of a taxonomy-free interest-aware recommendation system. The experimental setup involved a series of recommendation cycles, wherein users provided feedback on song suggestions produced by either TASTES and SRSE, which extract new *Spotify* at every cycle to better adapt to the user's preferences. A preliminary testing phase helped refine our methods, followed by a comprehensive 30-day user study with 20 participants, each exposed to both recommendation strategies.

Throughout the evaluation, we collected three main types of data: playlist audio composition, individual recommendation appraisals explicitly attributed by the users, and participant feedback via surveys pertaining to the music discovery experience achieved by each of the tested strategies. These data points allowed us to quantitatively and qualitatively compare the performance of TASTES and SRSE, providing a holistic view of their impact on music discovery, interest expansion and overall user satisfaction. The structured user study and systematic data collection methods provide a solid foundation for analysing and interpreting our findings.

In the next chapter, we delve into the results obtained through the evaluation study. We begin by providing a detailed description of our participant sample, followed by a comprehensive analysis of the data collected. We then draw the conclusions reflected by the results obtained in regards to the efficacy of TASTES and its potential advantages over traditional recommendation systems, with a focus on addressing the research questions that motivated this project.

5

Results

Contents

In this chapter, we present and analyse the results of the TASTES evaluation study. We begin by describing our participant sample, highlighting their demographics, music listening habits, *Spotify* usage patterns, and characteristics of their personal playlists. Subsequently, we detail the different evaluation methodologies employed in our analysis and the resulting findings. We finish by briefly discussing the obtained results in light of the research questions initially posed in this project.

5.1 Participant Sample

A total of 20 candidates were selected to undergo the evaluation study. All participants were *Spotify* users with a *Premium* subscription, a requirement for data collection through SWAPI, and at least one year of active usage on the platform. The following data was collected through the initial profiling questionnaire, detailed in Section 4.3.3.C, filled out by all participants at the start of the study.

5.1.1 Demographics

In Figure 5.1 and Figure 5.2 we present a breakdown of our participant pool by age and gender, respectively. As demonstrated by the graphs, the study group was composed of users between the ages of 21 and 26, consisting of 40% females and 60% male. These demographics align with *Spotify*'s predominant user base [41], which displays an identical gender distribution, with 55% of users falling within the 18-35 age bracket.

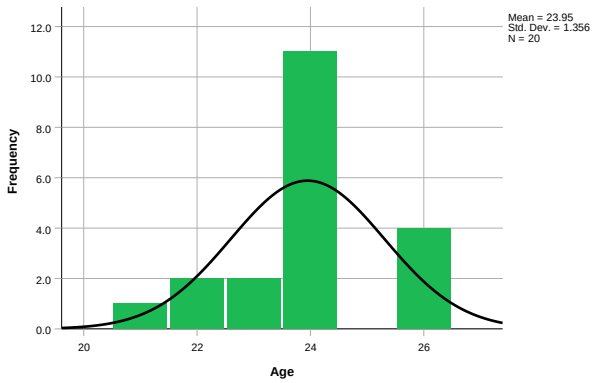


Figure 5.1: Participant Age Distribution [ID1]

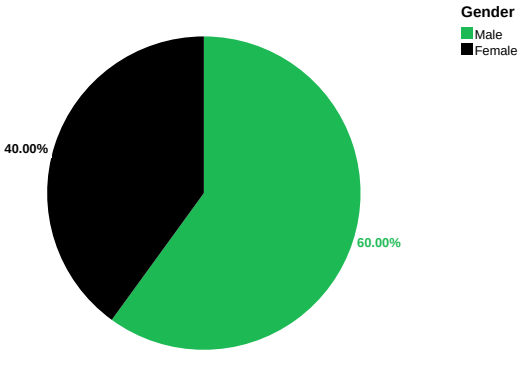


Figure 5.2: Participant Gender Distribution [ID2]

5.1.2 Musical Preference

Regarding music listening habits, the survey data shows most of the participant pool regularly integrates music into their daily lives, reflected by the answer distribution for question IM1, in Figure 5.3. The results also reflect a close-to-even distribution in terms of alignment with mainstream musical trends, with only 30% of the participants stating they are always sure of the music genre they wish to listen to. Furthermore, 50% of the candidates claim to prefer listening to music they already know, while only 20% favour music they have not heard before, and the remaining 30% show no preference either way. Concerning music discovery, 75% of the group stated they do not get tired of listening to new music, with none expressing the opposite view, as illustrated by the answer distribution for question IM4.

In addition to general listening trends, participants were also questioned on their preferences regarding music genre diversity in playlists. As evidenced in Figure 5.8, the candidate pool displays varying levels of affinity towards this particular playlist characteristic. Nevertheless, the group's majority (55%) seems to favour homogeneity over diversity, while only 25% states the opposite.

5.1.3 *Spotify* Usage

The participants were also inquired about select aspects of their *Spotify* usage. Almost all of the study sample has been an active *Spotify* user for at least 5 years, as illustrated in Figure 5.4, with a total

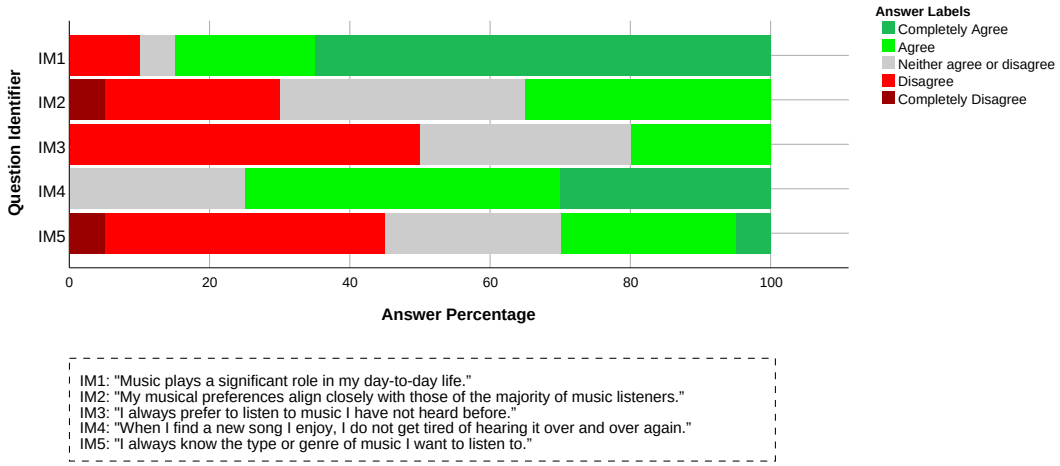


Figure 5.3: Survey Answer Distribution - Music Preferences [IM1-5]

group average of 7.8 years. In regards to frequency of usage, outlined in Figure 5.5, 75% of the group claims to spend on average between 5 to 20 hours a week using the streaming platform, coming to an overall mean of 12.8 hours per week. This also aligns with *Spotify*'s current usage statistics which show users based in North America spend an average of 140 minutes using the application every day [41], equivalent to about 16 hours per week. Finally, in response to question IS2, concerning the user's level of familiarity with the *Spotify* application and its various features, 18 of the 20 participants attributed at least a 4 out of 5.

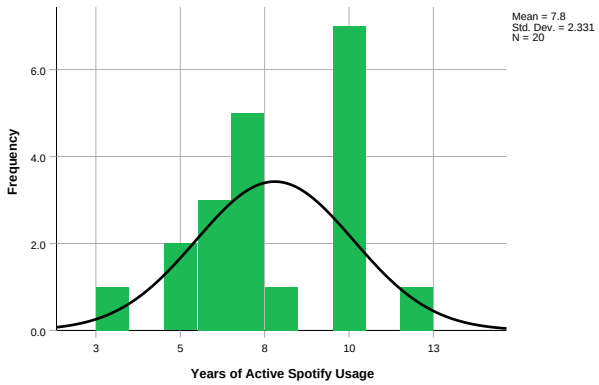


Figure 5.4: Active *Spotify* Usage Distribution [IS1]

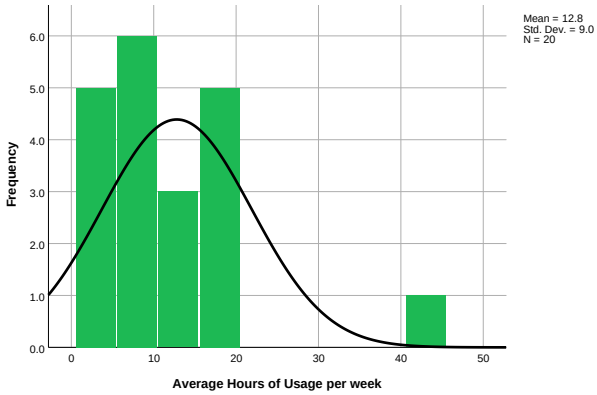


Figure 5.5: Spotify Usage per Week Distribution [IS2]

Participants were asked about their engagement with their personal playlists, including time spent listening to them and their approach to adding new tracks. As shown in Figure 5.6, 75% of the group chooses to spend at least 75% of their time on *Spotify* listening to personal playlists. Despite this, the distribution of answers to question IS5, represented in Figure 5.7, shows half the participants only add new tracks to their playlists weekly, with another 30% claiming this is only a monthly occurrence.

Lastly, the participants were questioned about their general satisfaction with *Spotify*'s recommenda-

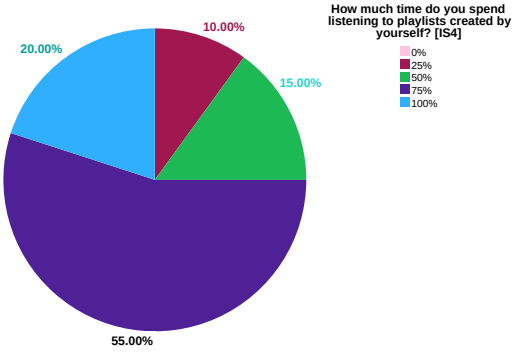


Figure 5.6: Survey Answer Distribution - Time Spent Listening to Self-Curated Playlists [IS4]

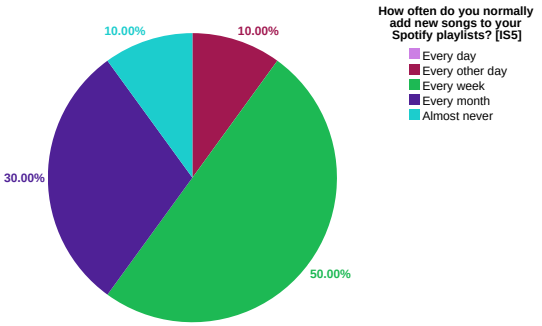


Figure 5.7: Survey Answer Distribution - Frequency of Additions to Personal Playlists [IS5]

tion capabilities. The results are depicted in Figure 5.9. The graph shows a mean score of 3.6 out of 5, with only 3 of the 20 participants attributing the maximum rating to *Spotify*'s ability to recommend relevant songs consistently.

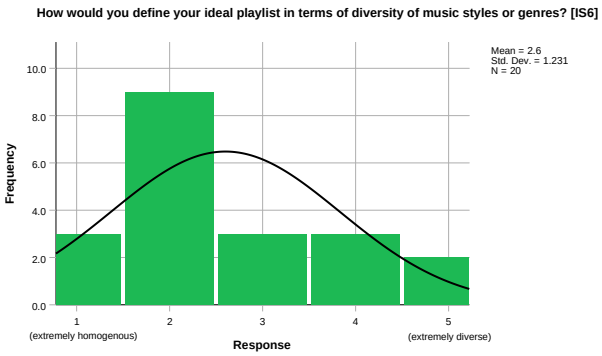


Figure 5.8: Survey Answer Distribution - Playlist Genre Diversity Preferences [IS6]

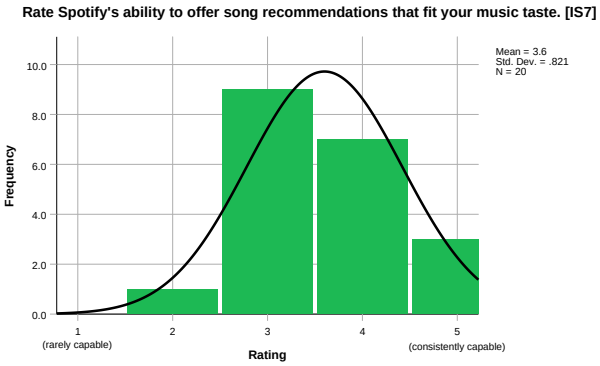


Figure 5.9: Survey Answer Distribution - Rating of *Spotify*'s Recommendation Capabilities [IS7]

5.1.4 Personal Playlist

Given our sample is also characterised by the curated playlists selected by the participants for the study, we also analysed some key characteristics of the initial playlist pool. Depicted in Figure 5.10, is a breakdown of the playlists by the number of tracks they contained at the start of the study. The distribution shows that the initial playlist pool was divided into 3 major groups according to size: under 50 tracks (35%), between 51 to 150 tracks (30%), and over 151 tracks (35%). At the start of the study, the shortest playlist was composed of 22 tracks while the largest was 2231 tracks long. Figure 5.12 and Figure 5.12 show the distributions of the number and size, i.e. number of tracks, of facets detected by TASTES' unsupervised classification method in our playlist pool. In regards to facet count, the sample shows a mean of 6.4 with a median of 4 facets found in half of the playlists. As for the number of tracks

comprising the facets, the sample displays a mean of 46.27 songs per facet with a median of 11.

Before being selected for the study, each participant was inquired regarding the overarching genre of music encompassed by their playlists, with the purpose of securing a sample which wasn't too homogeneous in this aspect. Figure 5.11 shows the distribution of the user's stated genres for their playlists. The 20 playlists were divided into 4 musical categories: Electronic, Hip-Hop, Rock, and Indie Pop. The most frequent genres found were Electronic, with 40% of the playlist population, and Hip-Hop, representing another 35%.

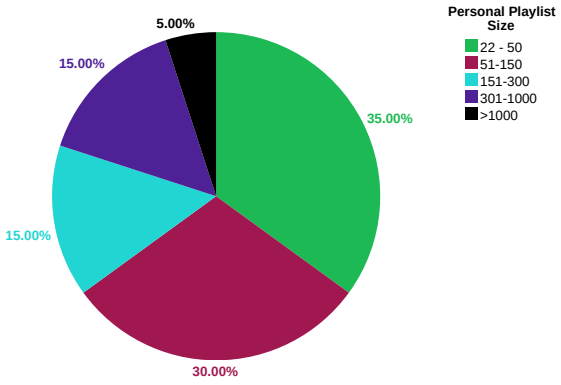


Figure 5.10: Personal Playlist Size Distribution

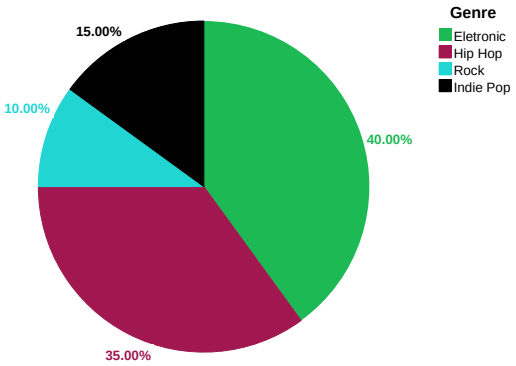


Figure 5.11: Personal Playlist Genre Distribution

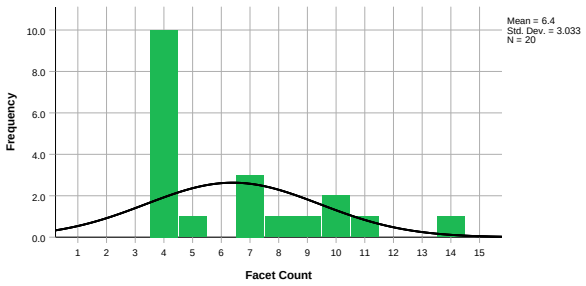


Figure 5.12: Personal Playlist Facet Count Distribution

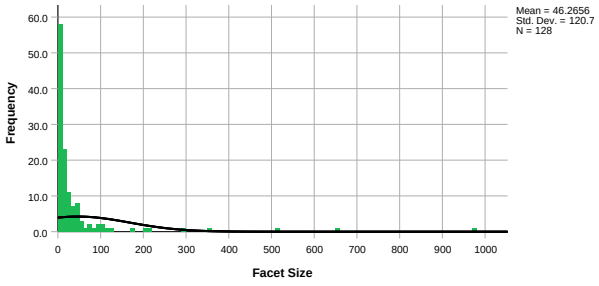


Figure 5.13: Personal Playlist Facet Size Distribution

In addition to this analysis, we also queried participants regarding their personal playlists. Figure 5.14 and Figure 5.15 respectively show the distributions of the group's answers when asked to rate their personal playlists on the diversity of music genres present and on their perceived repetitiveness. Both distributions demonstrate a diverse range of responses with a mean close to 5. In regards to the perceived diversity of the personal playlist, there is a slight inclination toward homogeneity, with a group median score of 4 for question IP1. As for perceived monotony, the sample shows an almost equally slight tendency towards seeing their playlists as more repetitive, reflected in the median score of 6 for survey question IP2.

In contrast to the previous distributions, the results for question IP4, presented in Figure 5.17, show that all participants rated their playlists between 6 and 8 out of 9 in terms of their listening experience,

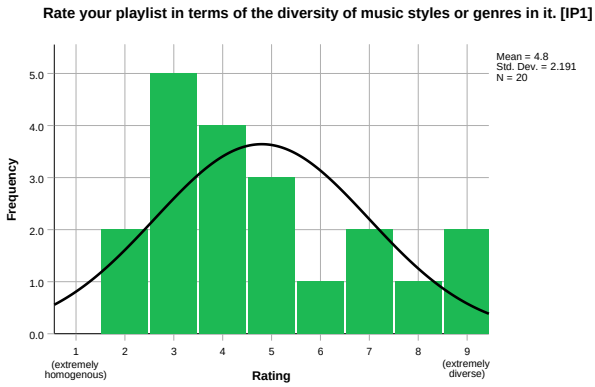


Figure 5.14: Survey Answer Distribution - Personal Playlist Diversity Rating [IP1]

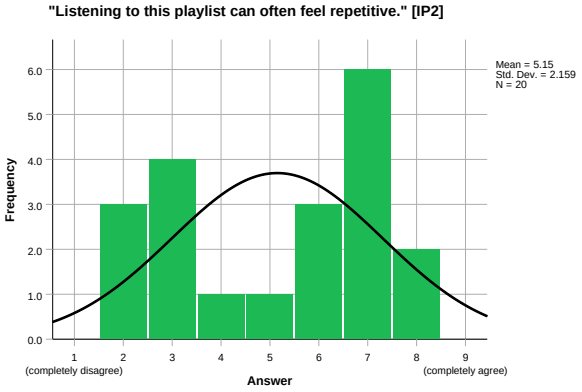


Figure 5.15: Survey Answer Distribution - Personal Playlist Repetitiveness [IP2]

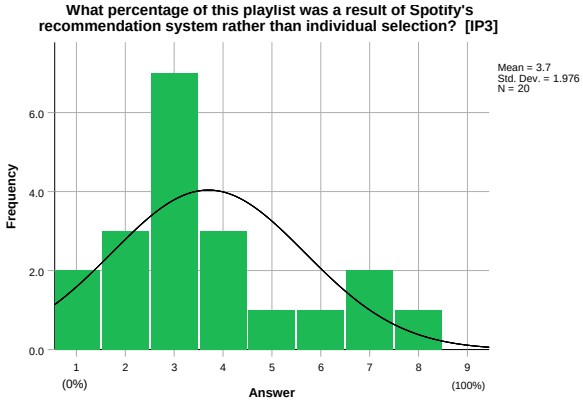


Figure 5.16: Survey Answer Distribution - Personal Playlist Recommendation Percentage [IP3]

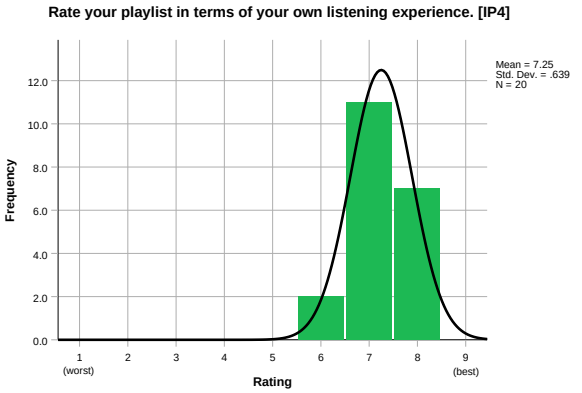


Figure 5.17: Survey Answer Distribution - Personal Playlist Experience Rating [IP4]

resulting in a dense distribution around the mean rating of 7.25. Finally, Figure 5.16 shows that most of the personal playlists selected by the participants were composed of a majority of hand-selected tracks, reflecting an approximate average of 25% of tracks originating from *Spotify* recommendations.

5.2 Results

In this section, we present the results of the evaluation study in light of their contribution to the initially posed research questions. For each of the underlying aspects of interest to our analysis, we introduce and justify the chosen quantitative and qualitative evaluation methodology and the statistical tests performed. Subsequently, we present and discuss the results obtained from each test.

All study participants scored at least a 4 out of 5 for questions FR_1S and FR_1T, pertaining to usage difficulties with the experimental setup during each testing segment. Consequently, we disregard any influence of user error in our analysis.

5.2.1 Musical Interest Expansion

To evaluate the first research question, we conducted quantitative data analysis by comparing the relative changes in the area of musical preferences covered in the personal playlists, as a result of each strategy. Additionally, we performed a qualitative data assessment focused on capturing the participants' perceived interest shifts after testing each recommendation approach.

5.2.1.A Quantitative Data Analysis

The quantitative assessment of the strategies' capacities for taste expansion was centred around two key aspects: the impact of each strategy on the playlists' audio feature distribution, informed by the changes to the features' variances, and the difference in the number of facet clusters detected by TASTES' unsupervised classification framework, before and after each testing segment.

Given the study sample's reduced size, we employed Wilcoxon's non-parametric signed-rank test [42] for the comparative analysis of playlist audio feature impact. Figure 5.18 and Figure 5.19 display the descriptive statistics and Wilcoxon's test results, respectively, for changes in audio feature variance caused by each approach. The data indicates that, on average, TASTES' suggestions led to a decrease in the variance of all playlist audio features. SRSE produced a similar effect, apart from *instrumentalness*, *loudness*, *key*, and *mode*, which show a slight average increase in variance as a result of employing *Spotify*'s methodology. Additionally, both strategies caused an average decrease in the number of facets identified by the unsupervised classification method, showing a mean change of -.75 and -1.05 after TASTES and SRSE respectively.

	N	Mean	Std. Deviation
Danceability variance change after SRSE	20	-.0009	.00189
Acousticness variance change after SRSE	20	-.0010	.00385
Energy variance change after SRSE	20	-.0010	.00385
Instrumentalness variance change after SRSE	20	.0006	.00410
Liveness variance change after SRSE	20	-.0008	.00368
Loudness variance change after SRSE	20	.1786	.95189
Speechiness variance change after SRSE	20	-.0002	.00094
Tempo variance change after SRSE	20	-11.3385	47.16878
Valence variance change after SRSE	20	-.0001	.00278
Time signature variance change after SRSE	20	-.0109	.02398
Mode variance change after SRSE	20	.0029	.01107
Key variance change after SRSE	20	.0175	.55829
Facet Count change after SRSE	20	-.7500	2.89964
Danceability variance change after TASTES	20	-.0011	.00168
Acousticness variance change after TASTES	20	-.0031	.00475
Energy variance change after TASTES	20	-.0031	.00475
Instrumentalness variance change after TASTES	20	-.0092	.03398
Liveness variance change after TASTES	20	-.0024	.00317
Loudness variance change after TASTES	20	-.7703	.88835
Speechiness variance change after TASTES	20	-.0006	.00065
Tempo variance change after TASTES	20	-45.1570	70.57712
Valence variance change after TASTES	20	-.0029	.00444
Time signature variance change after TASTES	20	-.0140	.04779
Mode variance change after TASTES	20	-.0028	.00855
Key variance change after TASTES	20	-.4594	.93121
Facet Count change after TASTES	20	-1.0500	3.47131

Figure 5.18: Descriptive Statistics - Audio Feature Variance Change

	Z	Asymp. Sig. (2-tailed)
Danceability variance change after TASTES - Danceability variance change after SRSE	-.933[b]	.351
Acousticness variance change after TASTES - Acousticness variance change after SRSE	-1.94[b]	.052
Energy variance change after TASTES - Energy variance change after SRSE	-1.94[b]	.052
Instrumentalness variance change after TASTES - Instrumentalness variance change after SRSE	-1.61[b]	.108
Liveness variance change after TASTES - Liveness variance change after SRSE	-2.39[b]	.017
Loudness variance change after TASTES - Loudness variance change after SRSE	-2.91[b]	.004
Speechiness variance change after TASTES - Speechiness variance change after SRSE	-2.02[b]	.044
Tempo variance change after TASTES - Tempo variance change after SRSE	-2.46[b]	.014
Valence variance change after TASTES - Valence variance change after SRSE	-2.69[b]	.007
Time signature variance change after TASTES - Time signature variance change after SRSE	-1.01[b]	.314
Mode variance change after TASTES - Mode variance change after SRSE	-.933[b]	.351
Key variance change after TASTES - Key variance change after SRSE	-1.12[b]	.263
Facet Count change after TASTES - Facet Count change after SRSE	-.270[c]	.788

b. Based on positive ranks.

c. Based on negative ranks.

Figure 5.19: Wilcoxon Signed-Rank Test Statistics - Audio Feature Variance Change

When it comes to the Wilcoxon's signed-ranks results, only 5 audio features seem to exhibit a significant statistical difference in the changes induced by both strategies: *liveness* ($Z = -2.389$, $p = .017$), *loudness* ($Z = -2.912$, $p = .004$), *speechiness* ($Z = -2.016$, $p = .044$), *tempo* ($Z = -2.464$, $p = .014$), and *valence* ($Z = -2.688$, $p = .007$). Although both approaches reduced the variance of most audio features in the personal playlists, TASTES appears to affect these five significantly more than SRSE. Regarding changes in facet count as a result of the suggestions produced, the data shows no significant difference between employing either strategy.

5.2.1.B Qualitative Data Analysis

To assess the strategies' perceived impact on interest expansion, the participants were inquired about four core aspects after experiencing each of the tested approaches:

- The overall diversity of the participant's personal playlist (survey question FP_4).
- The strategy's ability to produce recommendations outside of the music interests represented in the personal playlist (survey question FR_4).
- The accuracy of recommendations outside of the music interests represented in the personal playlist (survey question FR_5).
- The overall diversity of the produced recommendations (survey question FR_7).

The obtained results indicate no significant correlation between the responses given regarding SRSE and TASTES, as depicted in Figure 5.20. However, in regards to the overall diversity of the generated track suggestions, reflected by the data for question FR_7, SRSE comes close to showing notable improvement over TASTES, with a *p* value of .052 and a greater mean of 6.5 out of 9, compared to TASTES' resulting mean of 5.3. Ultimately, both approaches seem to produce identical results concerning the perceived diversity of the track picks produced as well as interest expansion as a result of the achieved PC experience.

	Z	Asymp. Sig. (2-tailed)
[FP_4T] Rate the level of diversity of your whole playlist by considering the variety of music styles and genres it contains at this point. (TASTES) - [FP_4S] Rate the level of diversity of your whole playlist by considering the variety of music styles and genres it contains at this point. (SRSE)	-.162[b]	.871
[FR_4T] "TASTES recommended music belonging to styles and genres not present in my playlist." (TASTES) - [FR_4S] "TASTES recommended music belonging to styles and genres not present in my playlist." (SRSE)	-1.342[c]	.180
[FR_5T] "I found TASTES' recommendations, belonging to music styles and genres not present in my playlist, interesting." (TASTES) - [FR_5S] "I found TASTES' recommendations, belonging to music styles and genres not present in my playlist, interesting." (SRSE)	-.600[c]	.549
[FR_7T] Rate the level of diversity of the radio recommendations produced by this version TASTES, considering the variety of music styles and genres represented. (TASTES) - [FR_7S] Rate the level of diversity of the radio recommendations produced by this version TASTES, considering the variety of music styles and genres represented. (SRSE)	-1.943[c]	.052

b. Based on negative ranks.

c. Based on positive ranks.

Figure 5.20: Wilcoxon Signed-Rank Test Statistics - Perceived Interest Expansion

5.2.2 Music Discovery Experience

In regards to the second posed research question, our quantitative data analysis was informed by the relative amount of feedback responses, i.e. ”likes”, additions, and rejections, attributed to the track suggestions produced by each methodology tested. Furthermore, we looked at the participants' impressions concerning different qualities of the music discovery journey experienced during each testing segment, collected through the feedback questionnaires introduced in Section 4.3.3.C, to evaluate this aspect from a qualitative framework.

5.2.2.A Quantitative Data Analysis

In line with the accuracy metric used by Kaya and Bridge [26], the quantitative evaluation of SRSE and TASTES' relative effectiveness for the PC task was guided by the number of relevant tracks produced in the recommendation sets. In our experimental setting, a suggestion is considered relevant if the user attributes a positive reaction, i.e. a ”like” or adding the track to their playlist. Consequently, in line with the comparative approach employed so far, a Wilcoxon's signed-rank test was conducted using the total count of added, liked, and rejected recommendations generated using both methodologies, as well as the percentage of songs added to the personal playlist which originated from the radio playlist. The results of the test are presented in Figure 5.21.

	Z	Asymp. Sig. (2-tailed)
[N_ADDED_TOTAL_T] Added recommendations total (TASTES) - [N_ADDED_TOTAL_S] Added recommendations total (SRSE)	-1.851[b]	.064
[ADDED_FACTOR_T] Percentage of added songs from radio (TASTES) - [ADDED_FACTOR_S] Percentage of added songs from radio (SRSE)	-.065[b]	.948
[N_LIKED_TOTAL_T] Liked recommendations total (TASTES) - [N_LIKED_TOTAL_S] Liked recommendations total (SRSE)	-1.708[c]	.088
[N_REJ_TOTAL_T] Rejected recommendations total (TASTES) - [N_REJ_TOTAL_S] Rejected recommendations total (SRSE)	-.853[b]	.393

b. Based on negative ranks.

c. Based on positive ranks.

Figure 5.21: Wilcoxon Signed-Rank Test Statistics - Positive Recommendation Feedback

As evidenced by the presented data, there were no significant differences found between the interactions generated by the methodologies. Although TASTES produced a higher mean of 14.8 total added recommendations comparatively to SRSE's mean of 12.7, the p value is still not small enough to be considered a relevant improvement. Consequently, both approaches seem to be equally effective in regard to the quantitative music discovery quality metrics chosen.

5.2.2.B Qualitative Data Analysis

The data used to evaluate TASTES' noticeable impact on music discovery tasks on *Spotify* from the user's perspective was collected through the feedback questionnaires filled out by candidates after each 15-day segment of the study. The group was inquired about their level of satisfaction with the same relevant aspects pertaining to both experiences:

- Level of repetitiveness induced by listening to the personal playlist (survey question FP_5).
- Personal playlist overall rating (survey question FP_6).
- Genre coverage of the presented suggestion, inside the playlist (survey question FR_2).
- Accuracy of the recommendations belonging to genres/themes already present in the personal playlist (survey question FR_3).
- Accuracy of the recommendations belonging to genres/themes not present in the personal playlist (survey question FR_5).
- Satisfaction with the overall PC experience provided (survey question FR_8).

Using the responses to the referenced survey question, all of which were on a 1-9 Likert scale, a Wilcoxon's signed-rank test was performed. The test results, displayed in Figure 5.22, illustrate that when considering the entire group of participants, there were no significant differences found in the perceived satisfaction resulting from employing either strategy, for any of the aspects analysed.

[FP_5T] "Currently, listening to my playlist evokes a sense of monotony or repetition." (TASTES) - [FP_5S] "Currently, listening to my playlist evokes a sense of monotony or repetition." (SRSE)	-.545[c]	.586
[FP_6T] Rate the playlist, at this point, in terms of your own listening experience. (TASTES) - [FP_6S] Rate the playlist, at this point, in terms of your own listening experience. (SRSE)	-.264[c]	.792
[FR_2T] "TASTES' recommendations covered all the music styles and genres present in my playlist." (TASTES) - [FR_2S] "TASTES' recommendations covered all the music styles and genres present in my playlist." (SRSE)	-.661[c]	.509
[FR_3T] "I found TASTES' recommendations, belonging to music styles and genres already present in my playlist, interesting." (TASTES) - [FR_3S] "I found TASTES' recommendations, belonging to music styles and genres already present in my playlist, interesting." (SRSE)	-1.164[c]	.244
[FR_5T] "I found TASTES' recommendations, belonging to music styles and genres not present in my playlist, interesting." (TASTES) - [FR_5S] "I found TASTES' recommendations, belonging to music styles and genres not present in my playlist, interesting." (SRSE)	-.600	.549
[FR_8T] Rate this version of TASTES in terms of your overall satisfaction with the recommended songs. (TASTES) - [FR_8S] Rate this version of TASTES in terms of your overall satisfaction with the recommended songs. (SRSE)	-.720[c]	.471

c. Based on negative ranks.

Figure 5.22: Wilcoxon Signed-Rank Test Statistics- Music Discovery Satisfaction (All Participants)

Upon obtaining these results, further research was conducted to understand TASTES' relative performance for different types of playlists. To do so, we calculated the Spearman rank-order correlation coefficients [43] between the differences in the responses provided by the participants to the same study questions, concerning each of the tested strategies, with the initial audio feature variances of their respective playlists. The objective of this subsequent analysis was twofold. Firstly, to look for correlations between the answers given to the survey questions concerning different aspects of music discovery,

thereby indicating if any of these are linked from the user's perspective. Additionally, to study the influence of personal playlist audio composition on the participants' relative satisfaction with the music discovery experience provided by SRSE and TASTES. The results of the Spearman rank-order correlation analysis can be observed in Figure E.1. We note that the variables representing the difference in responses to a certain query were named by adding the suffix "DIFF" to the corresponding question ID. For instance, FP_8_DIFF represents the difference between the candidates' overall satisfaction rating of the experiences provided by TASTES (FP_8T) and by SRSE (FP_8S).

In regards to the music discovery satisfaction inquiries, the results reveal four relevant correlations:

- `FR_2_DIFF` and `FR_3_DIFF`: The relevancy of recommendations belonging to genres already present in the personal playlist is positively correlated with the strategy's capacity to cover all different genres represented in the playlist (Spearman $rho(18) = .553$, $p = .011$).

- `FP_5_DIFF` and `FR_5_DIFF`: The relevancy of recommendations belonging to genres outside the personal playlist is negatively correlated with the perceived overall monotony of the playlist (Spearman $rho(18) = -.532$, $p = .016$).

- `FP_6_DIFF` and `FR_2_DIFF`/`FR_3_DIFF`: Participants' rating of their own playlists is positively correlated with the strategy's capacity to cover all different genres represented in the playlist (Spearman $rho(18) = .499$, $p = .025$), as well as with the relevancy of the recommendations belonging to those genres (Spearman $rho(18) = .470$, $p = .036$).

- `FP_8_DIFF` and `FR_3_DIFF`: The user's satisfaction with the music discovery experience provided by the recommendation strategy is heavily correlated with the relevancy of recommendations produced, belonging to genres already present in the personal playlist (Spearman $rho(18) = .666$, $p = .001$).

Concerning the correlations between the playlists' audio feature distributions and the user's discovery experience satisfaction, only one correlation was found to be relevant: FP_8_DIFF and VAR_INSTR_0 (Spearman $rho(18) = .601$, $p = .005$). Consequently, the user's overall satisfaction with the music discovery experience provided by TASTES over the one produced through SRSE is heavily correlated with the variance of *instrumentalness*, whose definition is included in Table A.1, in their playlist's tracks at the start of the study. When analysing the playlist sample according to *instrumentalness* variance, we found that 8 of the 20 playlists used in the study show a considerably lower value for this particular feature, relative to the rest of the sample, as illustrated in Figure 5.23.

To further study the relation between playlist *instrumentalness* variance and the success of TASTES for the task of its continuation, we conducted a Mann-Whitney U test [44] between two participant subgroups, informed by the distribution represented in Figure 5.23.

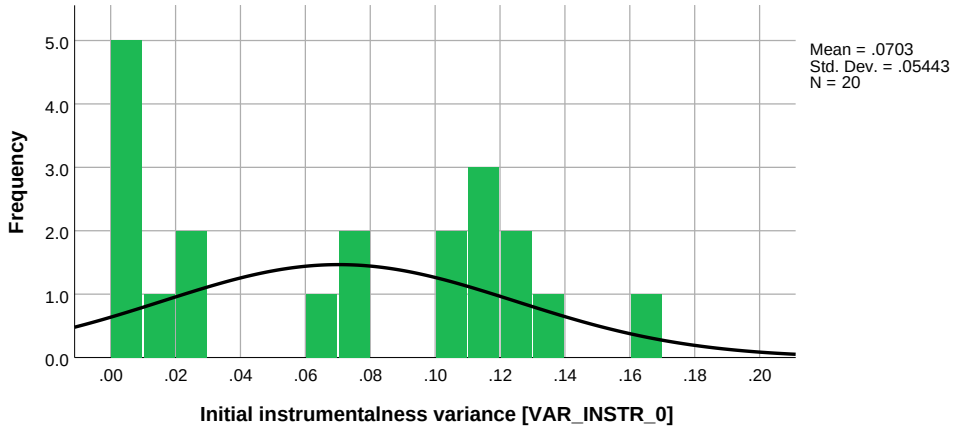


Figure 5.23: Initial Instrumentalness Variance Distribution of Personal Playlists

- Group A (8 participants): Participants whose personal playlists display an initial *instrumentalness* variance between the values of .00 to .03.
- Group B (12 participants): Participants whose personal playlists display an initial *instrumentalness* variance between the values of .06 to .20.

The results of the test are displayed in Figure E.2 and Figure E.3. Firstly, the data reveals that Group A's playlists have significantly higher *speechiness* variance than the rest of the sample ($U = 16$, $p = .014$). This aligns with the previously attained Spearman correlation data, illustrated in Figure E.1, which shows a significant negative correlation between *instrumentalness* and *speechiness* (Spearman $rho(18) = -.519$, $p = .019$). Moreover, Group A scores significantly higher in survey questions IM2 ($U = 17$, $p = .012$) and IM4 ($U = 22$, $p = .031$), reflecting interests that are more closely aligned with mainstream trends and showing to be less prone to getting tired of new songs, compared to Group B. In regards to the profiling queries concerning *Spotify* usage, there is only a significant difference in the sub-group responses for questions IS1 ($U = 14$, $p = .007$) and IS3 ($U = 22$, $p = .043$). Consequently, although Group A is composed of more recent *Spotify* users, they also claim to spend more time using the application per week relative to Group B. Lastly, Group B scores significantly higher in FP_3_DIFF ($U = 18.5$, $p = .017$) and FR_8_DIFF ($U = 7.5$, $p = .002$) than Group A. This indicates that for Group B, TASTES was not only significantly more apt than SRSE at suggesting songs belonging to genres included in their personal playlist but was also able to achieve a more satisfying music discovery experience overall. This is confirmed when running Wilcoxon's signed-rank test for Group B exclusively, using the answers to the Likert scale survey questions pertaining to music discovery. The results of this test are illustrated in Figure 5.24. They demonstrate that our proposed recommendation approach achieves significant improvements over SRSE for the same two music discovery aspects evidenced by the results of the Mann-Whitney analysis. These aspects are represented by survey questions FR_3 ($Z = 2.124$,

$p = .034$) and FR_8 ($Z = 2.546$, $p = .011$). Lastly, we note that Group A is composed of 6 Hip-Hop playlists, 1 Electronic, and 1 Indie Pop.

	Z	Asymp. Sig. (2-tailed)
[FP_4T] Rate the level of diversity of your whole playlist by considering the variety of music styles and genres it contains at this point. (TASTES) - [FP_4S] Rate the level of diversity of your whole playlist by considering the variety of music styles and genres it contains at this point. (SRSE)	-1.294[b]	.196
[FP_5T] "Currently, listening to my playlist evokes a sense of monotony or repetition." (TASTES) - [FP_5S] "Currently, listening to my playlist evokes a sense of monotony or repetition." (SRSE)	-.333[c]	.739
[FP_6T] Rate the playlist, at this point, in terms of your own listening experience. (TASTES) - [FP_6S] Rate the playlist, at this point, in terms of your own listening experience. (SRSE)	-.713[b]	.476
[FR_2T] "TASTES' recommendations covered all the music styles and genres present in my playlist." (TASTES) - [FR_2S] "TASTES' recommendations covered all the music styles and genres present in my playlist." (SRSE)	-1.469[b]	.142
[FR_3T] "I found TASTES' recommendations, belonging to music styles and genres already present in my playlist, interesting." (TASTES) - [FR_3S] "I found TASTES' recommendations, belonging to music styles and genres already present in my playlist, interesting." (SRSE)	-2.124[b]	.034
[FR_4T] "TASTES recommended music belonging to styles and genres not present in my playlist." (TASTES) - [FR_4S] "TASTES recommended music belonging to styles and genres not present in my playlist." (SRSE)	-1.904[c]	.057
[FR_5T] "I found TASTES' recommendations, belonging to music styles and genres not present in my playlist, interesting." (TASTES) - [FR_5S] "I found TASTES' recommendations, belonging to music styles and genres not present in my playlist, interesting." (SRSE)	-.052[b]	.959
[FR_7T] Rate the level of diversity of the radio recommendations produced by this version TASTES, considering the variety of music styles and genres represented. (TASTES) - [FR_7S] Rate the level of diversity of the radio recommendations produced by this version TASTES, considering the variety of music styles and genres represented. (SRSE)	-1.898[c]	.058
[FR_8T] Rate this version of TASTES in terms of your overall satisfaction with the recommended songs. (TASTES) - [FR_8S] Rate this version of TASTES in terms of your overall satisfaction with the recommended songs. (SRSE)	-2.546[b]	.011

b. Based on negative ranks.

c. Based on positive ranks.

Figure 5.24: Wilcoxon Signed-Rank Test Statistics- Music Discovery Satisfaction (Group A)

5.3 Discussion

As mentioned previously, a total of 20 candidates were selected to participate in the TASTES evaluation study. All participants were *Spotify* users with a *Premium* subscription and at least one year of active usage of the application. The study group consisted of users between the ages of 21 and 26, with 40% females and 60% males, in line with *Spotify*'s predominant user base. Almost all participants regularly incorporate music into their lives, showing mixed alignment with mainstream musical trends, with 75% of the group claiming not to get easily tired of listening to new music. Additionally, 80% of the sample does not favour listening to new music over music they already know, with 50% expressing the opposite view. These results could indicate a diminished utility of an expansion tool such as TASTES, whose objective is to constantly introduce new music to the user, for the majority of the participants, consequently restraining the results produced through the strategy.

Participants mainly spend between 5 to 20 hours a week using *Spotify*, mostly listening to personally curated playlists. Half of the candidates only add new tracks to their playlists weekly, with another 30% stating this is a monthly occurrence. This particular aspect may have also hindered the results obtained during the study, given our evaluation setup required the participants to add tracks to their playlists every 3 days, which is more frequent than 80% of the sample's usual behaviour. On average, participants scored *Spotify*'s recommendation capabilities around a 3.6 out of 5. The personal playlists were divided into 4 genres: 8 Electronic, 7 Hip-Hop, 3 Indie Pop, and 2 Rock. They varied in size, with 80% of them falling into the 0-200 tracks bracket. Although the majority of candidates favour homogeneity over diversity for the range of music genres in playlists, their own collections widely vary in this regard.

Despite this, all participants rated their personal playlists highly in terms of the listening experience they provide. The vast majority of the tracklists consisted of approximately 25% songs sourced from *Spotify* recommendations, while the remainder was hand-curated by the user. TASTES unsupervised clustering method detected an average of 6.2 facets per playlist, with half of the sample showing only 4 song categories. The average size of the facets found was 46.27 tracks, with close to 80% of the facets falling in the 0-50 song count bracket. Both these findings align with the results of Kaya and Bridge's sub-profile analysis [26], depicted in Figure 2.3 and Figure 2.4.

When evaluating our solution for the task of musical interest expansion, our quantitative analysis focused on changes in playlist audio feature variance and facet clusters detected by TASTES' classification method, resulting from each tested strategy. The results revealed that both the tested strategies result in a variance reduction for most of the audio features, with TASTES causing a more significant impact on *loudness*, *speechiness*, *tempo*, and *valence*. The qualitative data used for this assessment was provided by the participants as responses to *Likert* scale questions regarding different aspects of perceived recommendation diversity and interest changes resulting from the tested methodologies. No significant correlation emerged between the answers given for both strategies tested, suggesting TASTES shows no measurable improvement over *Spotify*'s recommendation approach in expanding the user's musical interests.

Concerning music discovery enhancement, we looked at the impact on the amount of positive and negative feedback produced by the participants, through interaction in the *Spotify* application, in response to individual track suggestions. The quantitative analysis revealed no significant differences between TASTES and SRSE, showing they are similarly apt at producing suggestions relevant to playlist continuation. The qualitative assessment of this aspect showed similar results. Based on comparative ratings provided by the participants through feedback surveys, of different aspects of their experience testing each strategy, no significant differences in user satisfaction were observed.

Further analysis explored correlations between playlist audio characteristics and user satisfaction. Notably, TASTES seemed to underperform for users with playlists characterised by a low *instrumentalness* variance, 80% of which belonged to the Hip-Hop genre. The data also reveals that *instrumentalness* and *speechiness* variance are negatively correlated. This aligns with *Spotify*'s description of the audio attributes, detailed in Table A.1. Accordingly, the playlists for which TASTES achieved a poorer performance are characterised by having tracks with varying levels of spoken word but similar levels of instrumental content. Furthermore, when juxtaposing this particular group of users with the rest of the participant sample, in regards to music consumption habits, they seem to report a stronger preference alignment with mainstream trends, as well as a higher weekly *Spotify* usage.

The particularities of the underperforming sub-group could be indicative of music consumption that is less concerned about audio similarity in songs and more dependent on other factors, such as lyricism,

messaging, personal affinity towards a certain artist, or even popularity. This is also supported by the fact that almost all underperforming playlists belong to the Hip-Hop genre, which is intrinsically characterised by spoken word and heavily commercialised in the current landscape, given its increasing popularity in recent years. Considering TASTES solely relies on *Spotify*'s audio features to interpret song similarity, the lack of accountability for these external factors could explain the poorer results for this particular group of playlists. Furthermore, given this sub-group reports a higher *Spotify* weekly usage, it is fair to assume they produce more relevant data to inform the streaming platform's preference elicitation methodologies. Accordingly, this could also have contributed to SRSE's improved performance for these cases.

When evaluating TASTES for the remaining participants, it showed significant improvement in user satisfaction compared to SRSE, particularly in generating relevant recommendations belonging to music genres already included in the personal playlists and enhancing the overall playlist continuation experience. TASTES' higher suggestion accuracy for musical themes included in the playlist is also in line with the results obtained from the audio feature variance change analysis. The results tell us that TASTES generated a more pronounced decrease in the variances, relative to SRSE, indicating it selects relevant suggestions which are more closely aligned with the initial audio feature composition of the participants' playlists.

In conclusion, the study has shown that TASTES does not demonstrate relevant improvement over *Spotify*'s recommendation approach as a musical interest expansion tool. However, the taxonomy-free methodology proved more capable than its counterpart at generating relevant suggestions inside the user's interest and providing a more satisfying PC experience overall, in cases where the level and content of speech are less relevant factors to the listener.

6

Conclusion

Contents

This chapter presents the final conclusions drawn from the totality of the research presented in this dissertation. We summarise the key findings from our research, discuss the limitations encountered during the study, and propose directions for future research to address these limitations and further explore the potential of our proposed recommendation strategy for enhancing music discovery tasks.

6.1 Summary of Findings

In this dissertation, we detail the development, implementation and evaluation of TASTES, a novel recommendation strategy built for *Spotify* playlist continuation and music interest expansion. The approach's design was inspired by the research conducted on the topics of adaptive music classification and interest-aware solutions to mitigating the accuracy-diversity tradeoff in recommendations. In light of our research findings, our solution leverages a taxonomy-free, interest-aware framework that dynam-

ically adapts to user preferences, aiming to provide relevant suggestions which expand their area of coverage in an iterative fashion. The strategy was evaluated through a 30-day within-subjects study with 20 *Spotify* users, comparing TASTES' performance in playlist continuation tasks to SRSE, a baseline strategy which emulates *Spotify*'s recommendation methodology by leveraging the services provided through their Web API. Participants were exposed to recommendations produced by both strategies and provided feedback on the selected suggestions through multiple recommendation cycles. The data collected for assessment included playlist audio composition changes, user-attributed individual recommendation appraisals, and satisfaction feedback provided through participant surveys. We summarise our findings in the vein of the primary research questions posed in this paper:

1. How effective is a taxonomy-free interest-aware recommendation approach such as TASTES at expanding the listener's musical interests?

2. Can a taste expansion approach to recommendation improve listeners' music discovery experience?

Regarding the first research aspect, TASTES exhibited no measurable improvement in expanding users' interests compared to the baseline strategy, demonstrating its interest-aware framework for recommendation offers no added capability for musical taste expansion over traditional taxonomy-dependent strategies. This was confirmed by both our quantitative evaluation, informed by the changes caused to the playlist's audio composition as a result of employing the recommendation strategy, and a qualitative assessment founded on the interest changes perceived by the study participants.

In regards to the second evaluated aspect, TASTES demonstrated improved results over *Spotify*'s methodology in cases where the level of spoken word is less relevant to the user and the level of instrumental content is more important. In these cases, the interest-aware approach proved to not only be more capable of generating relevant suggestions belonging to the user's existing interests but also providing a superior overall music discovery experience. This is evidenced by our qualitative assessment of this research aspect, informed by the study participant's feedback pertaining to the satisfaction with the recommendations produced through each strategy. We also note that when evaluated in this aspect for the totality of the study sample, TASTES showed matching results to *Spotify*'s recommendation approach.

The totality of the results obtained seems to indicate TASTES is a recommendation tool more apt for exploration than for expansion, particularly in cases where instrumental likeness is a more significant factor to the listener. Lastly, the successful integration of our approach with the *Spotify* application, leveraging its existing functionalities to enhance music discovery, confirms the feasibility of integrating external recommendation systems within established platforms to evaluate their performance in a modern and realist environment.

6.2 Limitations

Despite the promising results, several limitations were identified:

- **Reduced sample size:** The evaluation study was conducted using a relatively small sample of participants which may not fully capture the diversity of *Spotify* users' preferences. Analogously, the approach's capabilities were only tested for a limited range of musical genres, ultimately limiting the generalisability of any resulting findings.

- **Reduced evaluation time frame:** The restricted time window available for evaluation may not have allowed for the full assessment of the system's long-term effectiveness and adaptability. A more comprehensive preliminary assessment would have allowed a better understanding of the strategy's behaviour with different hyper-parameter values and provided opportunities to adjust the strategy's architecture to maximise the strategy's performance. Additionally, a longer study period could have provided more thorough insights into the results produced by TASTES in a less controlled environment.

- **User Interaction requirements:** Both the proposed strategy and its evaluation relied on users to regularly produce feedback interactions during the course of the study. This experimental setup may have influenced user behaviour, as evidenced by the data collected in the initial survey, causing deviation from typical usage patterns.

- **Self-Reported Data:** Reliance on self-reported data introduces potential biases and inaccuracies, impacting the reliability of some of the evaluated metrics employed.

- **Reliance on audio feature analysis:** TASTES solely relies on audio feature analysis to interpret song similarity, leaving out intangible factors that influence musical preferences, such as lyricism, messaging, personal affinity towards a certain artist, or cultural impact. This reliance may limit the strategy's effectiveness in capturing the full spectrum of user preferences.

- **Data collection limitations:** The strategy's reliance on SWAPI's data collection services to estimate the user's recent trends imposes certain constraints, such as rate limits and restricted access to some user activity data, ultimately limiting TASTES preference elicitation capabilities.

6.3 Future Research

We now outline some potential directions for future research aiming to address the limitations identified in our work and further assess TASTES' potential for music discovery:

- **Parameter tuning:** Conducting more extensive experiments with different balancing hyper-parameter values to better understand their impact on the strategy's performance and ultimately optimise the processing pipeline.
- **Comprehensive long-term study:** Conducting a long-term study capable of capturing the system's capability to adapt to the user's preferences over time in a less controlled environment, where the users can operate in accordance with their typical usage behaviours. Furthermore, employing a larger and more diverse sample of *Spotify* users in the evaluation, to better assess the strategy's versatility and extract more generalised findings.
- **Enhanced data collection:** Incorporate more objective measures of user engagement and satisfaction such as listening duration and skip rates, to complement self-reported data and reduce biases.
- **Real-time adaptation:** Enhance preference adaptability by incorporating dynamic adjustments based on immediate user feedback, collected in real-time, instead of the sequential processing approach employed.
- **Hybrid integration:** Explore the potential of integrating TASTES with other strategies that account for intangible factors beyond audio features.

With these considerations, we hope to motivate further research into the capabilities of approaches like TASTES for music recommendation, as well as into innovative methodologies for expanding music listener's interests in the contemporary landscape.

Bibliography

[1] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender system application developments: A survey," *Decision Support Systems*, vol. 74, pp. 12–32, 2015.

[2] S. M. McNee, J. Riedl, and J. A. Konstan, "Being accurate is not enough: how accuracy metrics have hurt recommender systems," *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, pp. 1097–1101, 2006.

[3] M. Kunaver and T. Požrl, "Diversity in recommender systems – a survey," *Knowledge-Based Systems*, vol. 123, pp. 154–162, 2017.

[4] P. Cheng, S. Wang, J. Ma, J. Sun, and H. Xiong, "Learning to recommend accurate and diverse items," in *Proceedings of the 26th International Conference on World Wide Web*, 2017, pp. 183–192.

[5] T. Di Noia, V. C. Ostuni, J. Rosati, P. Tomeo, and E. Di Sciascio, "An analysis of users' propensity toward diversity in recommendations," in *Proceedings of the 8th ACM Conference on Recommender Systems*, ser. RecSys '14. New York, NY, USA: Association for Computing Machinery, 2014, pp. 285–288. [Online]. Available: https://doi.org/10.1145/2645710.2645774

[6] T. Schäfer and P. Sedlmeier, "What makes us like music? determinants of music preference." *Psychology of Aesthetics, Creativity, and the Arts*, vol. 4, no. 4, p. 223, 2010.

[7] Spotify. Spotify users have spent over 2.3 billion hours streaming discover weekly playlists since 2015. [Online]. Available: https://newsroom.spotify.com/2020-07-09/spotify-users-have-spent-over-2-3-billion-hours-streaming-discover-weekly-playlists-since-2015/

[8] ——. Web api. [Online]. Available: https://developer.spotify.com/documentation/web-api

[9] N. D. Krebbers, "Automatic categorization of electronic music genres," Master's thesis, Utrecht University, 2020.

[10] A. Nair, S. Pillai, G. S. Nair, and A. T, “Emotion based music playlist recommendation system using interactive chatbot,” in *2021 International Conference on Communication and Electronics Systems (ICCES)*, 2021, pp. 1767–1772.

[11] MLP Software, LLC. Companion for spotify. [Online]. Available: https://mlpsoftwarellc.com/CompanionForSpotify/

[12] Spotify. Web api rate limits. [Online]. Available: https://developer.spotify.com/documentation/web-api/concepts/rate-limits

[13] N. Scaringella, G. Zoia, and D. Mlynek, “Automatic genre classification of music content: a survey,” *IEEE Signal Processing Magazine*, vol. 23, pp. 133–141, 2006.

[14] F. Pachet, D. Cazaly *et al.*, “A taxonomy of musical genres.” in *RIAO*. Citeseer, 2000, pp. 1238–1245.

[15] G. Tzanetakis and P. R. Cook, “Musical genre classification of audio signals,” *IEEE Trans. Speech Audio Process.*, vol. 10, pp. 293–302, 2002.

[16] F. Gouyon, S. Dixon, E. Pampalk, and G. Widmer, “Evaluating rhythmic descriptors for musical genre classification,” 05 2004.

[17] T. Lidy and A. Rauber, “Evaluation of feature extractors and psycho-acoustic transformations for music genre classification,” in *International Society for Music Information Retrieval Conference*, 2005.

[18] A. Flexer, E. Pampalk, and G. Widmer, “Novelty detection based on spectral similarity of songs.” in *ISMIR*, 2005, pp. 260–263.

[19] N. Scaringella and G. Zoia, “On the modeling of time information for automatic genre recognition systems in audio signals.” in *ISMIR*, 2005, pp. 666–671.

[20] S. B. Kotsiantis, I. Zaharakis, P. Pintelas *et al.*, “Supervised machine learning: A review of classification techniques,” *Emerging artificial intelligence applications in computer engineering*, vol. 160, no. 1, pp. 3–24, 2007.

[21] E. Forgy, “Cluster analysis of multivariate data: Efficiency versus interpretability of classification,” *Biometrics*, vol. 21, no. 3, pp. 768–769, 1965.

[22] K. R. Shahapure and C. Nicholas, “Cluster quality analysis using silhouette score,” in *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*, 2020, pp. 747–748.

[23] C. D. Manning, *An introduction to information retrieval*. Cambridge university press, 2009.

[24] I. T. Jolliffe and J. Cadima, “Principal component analysis: a review and recent developments,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150202, 2016. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2015.0202

[25] K. Bradley and B. Smyth, “Improving recommendation diversity,” 2001. [Online]. Available: https://api.semanticscholar.org/CorpusID:11075976

[26] M. Kaya and D. Bridge, “Subprofile-aware diversification of recommendations,” *User Modeling and User-Adapted Interaction*, vol. 29, no. 3, pp. 661–700, apr 2019. [Online]. Available: https://doi.org/10.1007%2Fs11257-019-09235-6

[27] Castells, *Novelty and Diversity in Recommender Systems*. Boston, MA: Springer US, 2015, pp. 881–918. [Online]. Available: https://doi.org/10.1007/978-1-4899-7637-6_26

[28] S. Vargas, “Novelty and diversity enhancement and evaluation in recommender systems and information retrieval,” 07 2014.

[29] C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, “Improving recommendation lists through topic diversification,” 01 2005.

[30] S. Vargas and P. Castells, “Rank and relevance in novelty and diversity metrics for recommender systems,” in *Proceedings of the fifth ACM conference on Recommender systems*, 2011, pp. 109–116.

[31] X. Zhao, Z. Zhu, and J. Caverlee, “Rabbit holes and taste distortion: Distribution-aware recommendation with evolving interests,” in *Proceedings of the Web Conference 2021*. ACM, apr 2021. [Online]. Available: https://doi.org/10.1145%2F3442381.3450099

[32] M. Kaya, “Accurate and diverse recommendations using item-based subprofiles,” 05 2018.

[33] D. O’Callaghan, D. Greene, M. Conway, J. Carthy, and P. Cunningham, “Down the (white) rabbit hole: The extreme right and online recommender systems,” *Social Science Computer Review*, vol. 33, pp. 459 – 478, 2015.

[34] M. Brown, J. Bisbee, A. Lai, R. Bonneau, J. Nagler, and J. A. Tucker, “Echo chambers, rabbit holes, and algorithmic bias: How YouTube recommends content to real users,” *SSRN Electronic Journal*, 2022. [Online]. Available: https://doi.org/10.2139%2Fssrn.4114905

[35] M. Hardt, E. Price, and N. Srebro, “Equality of opportunity in supervised learning,” *ArXiv*, vol. abs/1610.02413, 2016.

[36] S. Kullback and R. A. Leibler, “On Information and Sufficiency,” *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79 – 86, 1951. [Online]. Available: https://doi.org/10.1214/aoms/1177729694

[37] F. M. Harper and J. A. Konstan, “The movielens datasets: History and context,” *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, dec 2015. [Online]. Available: https://doi.org/10.1145/2827872

[38] H. Steck, “Calibrated recommendations,” in *Proceedings of the 12th ACM Conference on Recommender Systems*, ser. RecSys ’18. New York, NY, USA: Association for Computing Machinery, 2018, p. 154–162. [Online]. Available: https://doi.org/10.1145/3240323.3240372

[39] M. Kaya and D. Bridge, “Automatic playlist continuation using subprofile-aware diversification,” in *Proceedings of the ACM Recommender Systems Challenge 2018*, ser. RecSys Challenge ’18. New York, NY, USA: Association for Computing Machinery, 2018. [Online]. Available: https://doi.org/10.1145/3267471.3267472

[40] C.-W. Chen, P. Lamere, M. Schedl, and H. Zamani, “Recsys challenge 2018: Automatic music playlist continuation,” in *Proceedings of the 12th ACM Conference on Recommender Systems*, ser. RecSys ’18. New York, NY, USA: Association for Computing Machinery, 2018, p. 527–528. [Online]. Available: https://doi.org/10.1145/3240323.3240342

[41] J. Shepherd. 23 essential spotify statistics you need to know in 2024. [Online]. Available: https://thesocialshepherd.com/blog/spotify-statistics

[42] F. Wilcoxon, S. Katti, R. A. Wilcox *et al.*, “Critical values and probability levels for the wilcoxon rank sum test and the wilcoxon signed rank test,” *Selected tables in mathematical statistics*, vol. 1, pp. 171–259, 1970.

[43] S. B. Lyerly, “The average spearman rank correlation coefficient,” *Psychometrika*, vol. 17, no. 4, pp. 421–428, 1952.

[44] H. B. Mann and D. R. Whitney, “On a test of whether one of two random variables is stochastically larger than the other,” *The Annals of Mathematical Statistics*, vol. 18, no. 1, pp. 50–60, 1947. [Online]. Available: http://www.jstor.org/stable/2236101

A

Spotify Audio Feature Descriptions

Name	Description	Value Range
Acousticness	A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.	[0-1]
Danceability	Danceability describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.	[0-1]
Energy	Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale.	[0-1]
Instrumentalness	Predicts whether a track contains no vocals. ”Ooh” and ”aah” sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly ”vocal”. The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0.	[0-1]
Liveness	Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.	[0-1]
Speechiness	Speechiness detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.	[0-1]
Valence	A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).	[0-1]
Loudness	The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. Loudness is the quality of a sound that is the primary psychological correlate of physical strength (amplitude). Values typically range between -60 and 0 dB.	[-60-0] (dB)
Tempo	The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration.	[0-∞] (BPM)
Key	The key the track is in. Integers map to pitches using standard Pitch Class notation. E.g. 0 = C, 1 = C$\sharp$/D$\flat$, 2 = D, and so on. If no key was detected, the value is -1.	[-1-11]
Time Signature	An estimated time signature. The time signature (meter) is a notational convention to specify how many beats are in each bar (or measure). The time signature ranges from 3 to 7 indicating time signatures of ”3/4”, to ”7/4”.	[3-7]
Mode	Mode indicates the modality (major or minor) of a track, the type of scale from which its melodic content is derived. Major is represented by 1 and minor is 0.	$\{0,1\}$

Table A.1: *Spotify* Audio Feature Descriptions and Value Ranges

B

Final Implementation Hiper-Parameter Values

Identifier	Description	Value
α_{recent}	Relevance Function Slope	1.0
β_{recent}	Relevance Function Mid-point	.50
MIN_F	Minimum Facet Count	4
MAX_F	Maximum Facet Count	15
η_{decay}	Relevance Decay Factor (Primary/Non-primary Tracks)	.95 / .90
ϕ_{stream}	Stream Count Factor	.50
ζ_{ss}	Seed Success Factor	.25
θ_{add}	Added Track Weight Factor	.80
$\theta_{success}$	Successful Seed Weight Factor	.70
θ_{liked}	Liked Track Weight Factor	.50
θ_{str}	Streamed Track Weight Factor	.30
θ_{model}	Model Track Weight Factor	.20
θ_{rej}	Rejected Track Count Factor	-.20
θ_{rem}	Removed Count Factor	-.40
γ	Preference Daily Update Factor	.05
λ	Continuation/Expansion Tradeoff Factor	.50
N_REQ	Recommendation Request Size	20
N_FINAL	Recommendation Request Size	15

Table B.1: Hiper-Parameter Values - TASTES Final Implementation

C

User Study Guidelines

TASTES User Study Guidelines

Introduction

Welcome to the **TASTES Evaluation Study**! In this study, you will be part of a select group of candidates evaluating the performance of TASTES at the user level. TASTES is a music recommendation algorithm designed to suggest songs based on user-curated playlists on Spotify. Its objective is to understand users' listening preferences within specific playlists, better adapt to their evolving preferences over time, and introduce them to new music that can expand their area of interest. The objective of the study is to evaluate two distinct configurations of TASTES, called **TASTES-A** and **TASTES-B**, and compare their impact on the Spotify user's listening experience.

Study Duration

The study will span **30 days**, from **March 16th (Saturday) to April 15th (Monday),** and is divided into two sections of 15 days each. During each of the sections the candidates will be testing either TASTES-A or TASTES-B. At the end of each section, they will be asked to answer a short questionnaire pertaining to their experience using TASTES during the previous 15-day period.

Testing Process

Personal Playlist Selection

Before the start of the testing period, each candidate has been asked to select a Spotify playlist, created and curated by themselves. Ideally, you should have chosen a playlist you listen to frequently (at least 3 times a week) and during the same activity such as running, working out, studying, driving, etc. This playlist will be referred to as your **"personal playlist"**.

Radio Playlist

Throughout the testing period, a designated playlist with the prefix "[RADIO]" and our cover logo will be available in each candidate's Spotify Library, as illustrated in Figure 1. The "**radio playlist**" will contain **15 song recommendations** specifically curated by TASTES for you to listen to and provide feedback on.

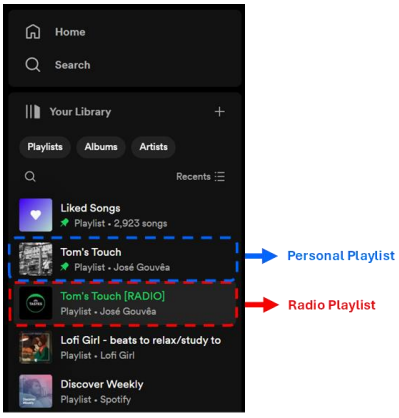


Figure 1: Personal Playlist and Corresponding Radio Playlist

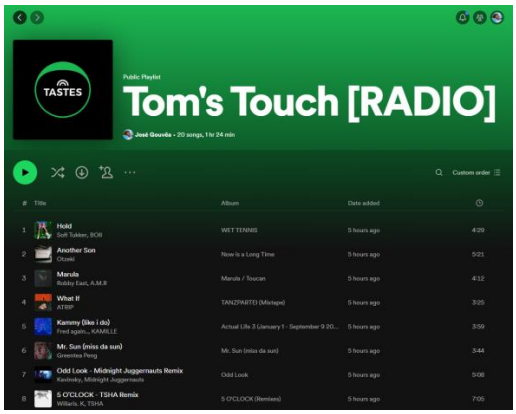


Figure 2: Radio Playlist

User Feedback

You can react to each song recommendation in your radio playlist in **3 different ways**, according to how you feel about that song in particular:

- Attributing a "**like**" to the song = "I like this song." (figure 3)
- **Adding** the song to your personal playlist = "This is the kind of recommendation I'm looking for." (figure 4)
- **Removing** the song from the radio playlist = "I don't want recommendations like this one." (figure 5)

If the song doesn't fit any of the above, simply **leave it** without a reaction. This will indicate that the recommendation was not sufficiently good or bad to deserve a positive or negative reaction from you. You are not required to listen to each song in its entirety, if you feel like you've listened to enough of a song to decide if and how you wish to react to it then you can do so and move on to the next recommendation.

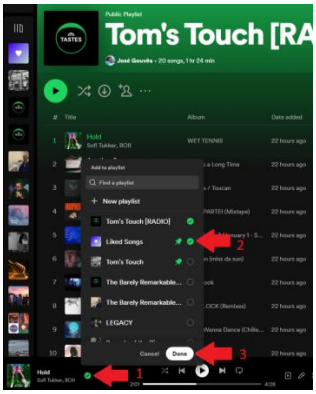


Figure 3: Liking a Song

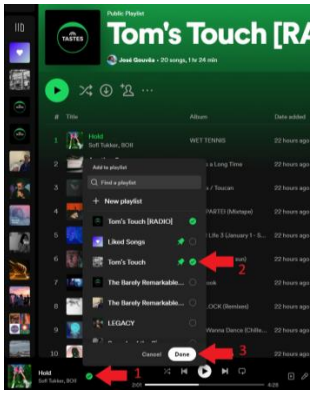


Figure 4: Adding a Song

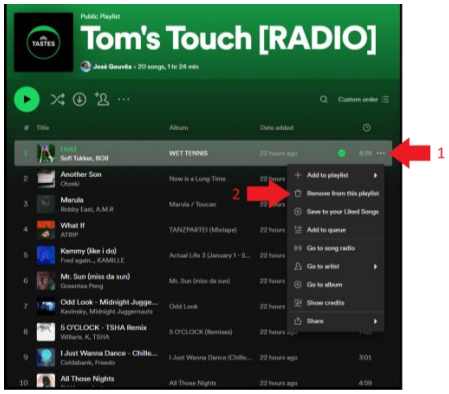


Figure 5: Removing a Song

Radio Refresh

The radio playlist will refresh automatically **every 3 days**, replacing the previous recommendations with 15 new ones. Candidates are asked to **react** to at least 3 of the 15 recommendations produced during each 3-day period. If you fail to produce the minimum number of reactions at the end of a 3-day mark, you will be reminded to do so via a WhatsApp message.

Further Considerations

TASTES tries to understand your music interests by analysing your listening history on Spotify. To facilitate this process, we encourage you to **listen to your personal playlist** as much as possible, along with any recommendations you've especially enjoyed. Moreover, any modifications made to your personal playlist, such as adding or removing songs, are also taken into account by TASTES. The more usage data you generate, the better TASTES will become at capturing and adapting to your preferences over time.

Finally, we ask that you **do not delete** either the personal or radio playlists at any point during the study, as it will compromise the testing process and thereby terminate your participation in the study.

Configuration Switch

After the first 15 days of testing, all candidates will switch the configuration of TASTES they are testing. This change does not impact your testing process; you should continue using Spotify the same way you did since the start of the study.

Feedback Questionnaire

At the end of each 15-day section, candidates will complete a short questionnaire providing feedback and relative satisfaction with the recommendations produced by TASTES-A and TASTES-B.

Further Clarification

For any further clarification regarding TASTES or any part of the study and testing process, candidates are encouraged to ask questions.

We greatly appreciate your time and participation in this study! Let's dive into the world of music discovery together! 🎶

D

User Study Surveys

TASTES Evaluation Study - Initial Survey

Welcome to the **TASTES Evaluation Study**!

TASTES is an approach to music recommendation designed to personalize song suggestions based on user-curated playlists on Spotify and enhance the playlist listening experience. The development of this project is part of a Master's Dissertation, from Instituto Superior Técnico.

If you choose to participate in this study, you will be part of a select group of candidates evaluating the performance of two different versions of TASTES. To assist us in characterizing our candidate demographics, we kindly request that you share your age and gender, along with any pertinent details regarding your listening habits as a music enthusiast and your user behavior on the Spotify platform.

The 30-day study will be conducted from **March 16th to April 15th** and is divided into **two 15-day segments**. A brief satisfaction survey will be administered after each segment.

Before proceeding to the survey, make sure you understand and consent to the following terms:

1 - You have read and understood the meaning of this study. You have the opportunity to ask questions, if necessary, and collect the respective answers.

2 - You understand that participation in this study is voluntary and that you can withdraw at any time without giving any explanation. If this happens, you will not be penalized and the data relating to your experiment will be removed and destroyed.

3 - You authorize the recording of your Spotify usage data such as likes, recent streaming history, and playlist edits, through the duration of the study. The data will be archived in a private server with restricted access and will be destroyed in accordance with the General Data Protection Regulation.

4 - You authorize the collection and processing of your anonymized Spotify usage data within the scope of this project for the purposes of analysis, investigation, and dissemination of results in scientific publications or conferences in the project area.

Thank you for your participation! For any questions, please contact jose.m.gouvea@tecnico.ulisboa.pt.

* Indicates required question

Identification

Before proceeding to the survey, please identify yourself with the TASTES study candidate ID provided along with the link to this survey.

1. Please introduce your candidate ID: *

Skip to question 2

Demographics

In this section, we kindly request that you share some fundamental demographic details.

2. **[ID1]** Gender *

Mark only one oval.

- Female
- Male
- Other: _____

3. **[ID2]** Age *

Music Listening Habits

To help us gain a better understanding of your listening habits as a music enthusiast, please indicate your level of agreement with the following statements:

11. **[IS3]** On average, how many hours a week do you spend listening to music on Spotify? *

12. **[IS4]** How much time do you spend listening to playlists created by yourself? *

Mark only one oval.

1 2 3 4 5

Non ◯ ◯ ◯ ◯ ◯ All the time (100% of the time)

13. **[IS5]** How often do you normally add new songs to your Spotify playlists? *

Mark only one oval.

- Almost never
- Every month
- Every week
- Every other day
- Every day

14. **[IS6]** How would you define your ideal playlist in terms of diversity of music styles or genres? *

Mark only one oval.

1 2 3 4 5

Extn ◯ ◯ ◯ ◯ ◯ Extremely diverse

15. **[IS7]** According to your own experience, rate Spotify's ability to offer song recommendations that fit your music taste. *

Mark only one oval.

1 2 3 4 5

Rare ◯ ◯ ◯ ◯ ◯ Consistently capable

Your Playlist

In this section, we inquire about the playlist you selected for the purpose of testing. If you forgot the playlist you selected, it is indicated along with your candidate ID before you were directed to this survey.

16. **[IP1]** Rate your playlist in terms of the diversity of music styles or genres in it: *

Mark only one oval.

1 2 3 4 5 6 7 8 9

Extn ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ Extremely diverse

4. **[IM1]** "Music plays a significant role in my day-to-day life." *

Mark only one oval.

1 2 3 4 5

Com ◯ ◯ ◯ ◯ ◯ Completely agree

5. **[IM2]** "My musical preferences align closely with those of the majority of music listeners." *

Mark only one oval.

1 2 3 4 5

Com ◯ ◯ ◯ ◯ ◯ Completely agree

6. **[IM3]** "I always prefer to listen to music I have not heard before." *

Mark only one oval.

1 2 3 4 5

Com ◯ ◯ ◯ ◯ ◯ Completely agree

7. **[IM4]** "When I find a new song I enjoy, I do not get tired of hearing it over and over again." *

Mark only one oval.

1 2 3 4 5

Com ◯ ◯ ◯ ◯ ◯ Completely agree

8. **[IM5]** "I always know the type or genre of music I want to listen to." *

Mark only one oval.

1 2 3 4 5

Com ◯ ◯ ◯ ◯ ◯ Completely agree

Spotify Usage

In this section, we would like to understand your experiences as a Spotify user.

9. **[IS1]** For how long have you been a Spotify user? *

10. **[IS2]** How familiar would you say you are with the Spotify application and its functionalities? *

Mark only one oval.

1 2 3 4 5

Extn ◯ ◯ ◯ ◯ ◯ Extremely familiar

17. **[IP2]** State your agreement with the following statement: *
"Listening to this playlist can often feel repetitive."

Mark only one oval.

1 2 3 4 5 6 7 8 9

Com ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ Completely agree

18. **[IP3]** What percentage of this playlist was a result of Spotify's recommendation system rather than individual selection? *

Mark only one oval.

1 2 3 4 5 6 7 8 9

0% (◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ 100% (all recommended)

19. **[IP4]** Rate the playlist in terms of your own listening experience.

Mark only one oval.

1 2 3 4 5 6 7 8 9

(wor ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ (best)

User Study Guidelines

You have completed the initial study survey! You can find all the necessary candidate guidelines for the period of testing that follow here. For any questions or clarifications, please contact jose.m.gouvea@tecnico.ulisboa.pt.

TASTES - User Feedback Survey

INTRO - END OF SEGMENT 1

Welcome back to the **TASTES Evaluation Study**!

The objective of the following survey is to collect participants' feedback after the first 15 days of the study, from **March 16th** to **March 31st**, and their experience with the first version of TASTES.

Please make sure you have listened and reacted to your last playlist radio recommendations, published on March 28th, **before** proceeding to this survey!

This survey is composed of two sections. The first section refers to your experience with your own personal playlist. The second section refers to your experience with the TASTES radio playlist, where the recommendations produced by TASTES have been published since the start of the study.

After completing this survey you will proceed to the **second half of the study**, where you will be testing the second version of TASTES for the following 15 days. We remind you that even though you will be testing a different version of TASTES, this change has no practical implications for the candidates' testing procedure. You should keep using Spotify the same way you have since the beginning of the study, keeping in mind you are now testing a different version of TASTES than before. The second half of the study concludes on April 15th, after which you will be asked to fill out the second and final feedback survey, pertaining to your experience with the second version of TASTES.

Thank you once more for your participation and availability! For any questions, please contact jose.m.gouvea@tecnico.ulisboa.pt.

INTRO - END OF SEGMENT 2

Welcome back to the **TASTES Evaluation Study**!

The objective of the following survey is to collect participants' feedback after the second 15-day period of the study, from **April 1st** to **April 16th**, and their experience with the second version of TASTES.

Please make sure you have listened and reacted to your last playlist radio recommendations, published on April 13th, **before** proceeding to this survey!

This survey is composed of two sections. The first section refers to your experience with your own perosnal playlist. The second section refers to your experience with the TASTES radio playlist, where the recommendations produced by TASTES are published.

This survey is identical to the first user feedback survey which was carried out at the end of the first half of this study. A record of some of your previous answers is provided in a pdf document along with the link to this survey, so you can take them into account when deciding on your answers this time around. Remember, the answers on the pdf refer to the first version of TASTES you tested, whilst the ones you provide now will refer to the second version.

After completing this survey, you will conclude the TASTES Evaluation Study and you are therefore no longer required to use Spotify as requested for the purposes of this study.

Thank you once again for your participation and availability! For any questions, please contact jose.m.gouvea@tecnico.ulisboa.pt.

* Indicates required question

Identification

Before proceeding to the survey, please identify yourself with the TASTES study candidate ID provided along with the link to this survey.

1. Please introduce your TASTES candidate ID: *

Your Personal Playlist

In this section, we inquire about your experience, **in the last 15 days,** with the playlist you initially selected **before** starting this study, also referred to as your **personal playlist**. The name of your personal playlist is indicated along with your candidate ID before you were directed to this survey.

2. **[FP_4]** Rate the level of diversity of your whole playlist by considering the variety of music styles and genres it contains at this point: *

Mark only one oval.

1 2 3 4 5 6 7 8 9

Extr ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ Extremely diverse

3. **[FP_5]** State your agreement with the following statement: *
"Currently, listening to my playlist evokes a sense of monotony or repetition."

Mark only one oval.

1 2 3 4 5 6 7 8 9

Con ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ Completely agree

4. **[FP_6]** Rate the playlist in terms of your own listening experience. *

Mark only one oval.

1 2 3 4 5 6 7 8 9

(wo ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ (best)

TASTES radio recommendations

In this section, we enquire about your experience with the song recommendations produced by TASTES for your playlist radio, **during the last 15 days**.

5. **[FR_1]** State your agreement with the following statement: *
"I had no trouble using the TASTES radio playlist on Spotify as it was instructed for the purposes of this study."

Mark only one oval.

1 2 3 4 5

Con ◯ ◯ ◯ ◯ ◯ Completely agree

6. **[FR_2]** State your agreement with the following statement: *
"TASTES' recommendations covered all the music styles and genres present in my playlist."

Mark only one oval.

1 2 3 4 5 6 7 8 9

Con ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ Completely agree

7. **[FR_3]** State your agreement with the following statement: *
"I found TASTES' recommendations, belonging to music styles and genres already present in my playlist, interesting."

Mark only one oval.

1 2 3 4 5 6 7 8 9

Con ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ Completely agree

8. **[FR_4]** State your agreement with the following statement: *
"TASTES recommended music belonging to styles and genres not present in my playlist."

Mark only one oval.

1 2 3 4 5 6 7 8 9

Con ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ Completely agree

9. **[FR_5]** State your agreement with the following statement: *
"I found TASTES' recommendations, belonging to music styles and genres not present in my playlist, interesting."

Mark only one oval.

1 2 3 4 5 6 7 8 9

Con ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ Completely agree

10. **[FR_7]** Rate the level of diversity of the radio recommendations produced by this version TASTES, considering the variety of music styles and genres represented: *

Mark only one oval.

1 2 3 4 5 6 7 8 9

Extn ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ Extremely diverse

11. **[FR_8]** Rate this version of TASTES in terms of your overall satisfaction with the recommended songs. *

Mark only one oval.

1 2 3 4 5 6 7 8 9

Extn ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ ◯ Extremely satisfied

Survey Completed!

OUTRO - END OF SEGMENT 1
You have completed the first feedback survey! We remind you again that from this point forward, once your radio playlist is refreshed on the **31st of March**, you will be testing a **different version** of TASTES.

If you need to be reminded of the candidate guidelines for testing, you can find them here. For any questions or clarifications, please contact jose.m.gouvea@tecnico.ulisboa.pt. Thank you and happy listening!

OUTRO - END OF SEGMENT 2

You have completed the second feedback survey and reached the end of the TASTES Evaluation Study!

You are free to delete the TASTES radio playlist from your Spotify Library and resume your normal usage of the Spotify application. Thank you for your availability, patience and your vital contribution to this project!

For any questions you may have reguarding the study, please contact jose.m.gouvea@tecnico.ulisboa.pt.

E

Statistical Analysis Results

Correlations

			[VAR_DANCE_0] Initial danceability variance	[VAR_ACUST_0] Initial accousticness variance	[VAR_ENERG_0] Initial energy variance	[VAR_INSTR_0] Initial instrumentalness variance	[VAR_LIVE_0] Initial liveness variance	[VAR_LOUD_0] Initial loudness variance	[VAR_SPCH_0] Initial speechiness variance	[VAR_TEMPO_0] Initial tempo variance	[VAR_VAL_0] Initial valence variance	[VAR_TSIG_0] Initial time signature variance	[VAR_MODE_0] Initial mode variance	[VAR_KEY_0] Initial key variance	FP_5_DIFF (FP_5T - FP_5S)	FP_6_DIFF (FP_6T - FP_6S)	FR_2_DIFF (FR_2T - FR_2S)	FR_3_DIFF (FR_3T - FR_3S)	FR_5_DIFF (FR_5T - FR_5S)	FR_8_DIFF (FR_8T - FR_8S)
Spearman's rho	[VAR_DANCE_0] Initial danceability variance	Correlation Coefficient	1.000	.423	.386	-.262	.230	.059	.251	.674**	.239	.605**	-.053	.006	-.094	-.141	-.064	-.156	-.008	-.179
		Sig. (2-tailed)	.	.063	.092	.265	.329	.806	.286	.001	.310	.005	.826	.980	.695	.553	.788	.513	.972	.449
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	[VAR_ACUST_0] Initial accousticness variance	Correlation Coefficient	.423	1.000	.747**	-.350	-.062	.323	-.005	.424	.480*	.434	.177	-.242	.112	-.171	-.369	.010	-.057	-.250
		Sig. (2-tailed)	.063	.	<.001	.130	.796	.164	.985	.062	.032	.056	.454	.304	.640	.470	.109	.966	.811	.289
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	[VAR_ENERG_0] Initial energy variance	Correlation Coefficient	.386	.747**	1.000	-.029	.030	.576**	-.353	.090	.496*	.223	-.080	-.077	.018	-.097	-.324	.048	-.036	.013
		Sig. (2-tailed)	.092	<.001	.	.905	.900	.008	.126	.705	.026	.345	.738	.748	.940	.684	.164	.841	.881	.957
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	[VAR_INSTR_0] Initial instrumentalness variance	Correlation Coefficient	-.262	-.350	-.029	1.000	-.257	.113	-.519*	-.268	.084	-.166	-.060	.564**	-.224	.245	.084	.348	.139	.601**
		Sig. (2-tailed)	.265	.130	.905	.	.274	.636	.019	.254	.724	.484	.801	.010	.343	.297	.723	.133	.558	.005
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	[VAR_LIVE_0] Initial liveness variance	Correlation Coefficient	.230	-.062	.030	-.257	1.000	-.433	-.030	.002	.379	.236	-.272	.033	.256	.079	.088	-.118	-.092	-.228
		Sig. (2-tailed)	.329	.796	.900	.274	.	.056	.900	.995	.099	.316	.246	.890	.276	.741	.714	.621	.699	.333
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	[VAR_LOUD_0] Initial loudness variance	Correlation Coefficient	.059	.323	.576**	.113	-.433	1.000	-.538*	-.239	.029	-.233	.059	-.110	-.111	-.252	-.322	.122	-.073	.268
		Sig. (2-tailed)	.806	.164	.008	.636	.056	.	.014	.310	.905	.322	.806	.645	.642	.285	.166	.609	.759	.254
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	[VAR_SPCH_0] Initial speechiness variance	Correlation Coefficient	.251	-.005	-.353	-.519*	-.030	-.538*	1.000	.505*	-.356	.282	.069	-.386	-.136	.042	-.020	-.338	.195	-.395
		Sig. (2-tailed)	.286	.985	.126	.019	.900	.014	.	.023	.123	.229	.772	.092	.568	.859	.932	.145	.410	.085
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	[VAR_TEMPO_0] Initial tempo variance	Correlation Coefficient	.674**	.424	.090	-.268	.002	-.239	.505*	1.000	.069	.782**	-.179	.077	-.098	-.050	-.146	-.186	.206	-.317
		Sig. (2-tailed)	.001	.062	.705	.254	.995	.310	.023	.	.772	<.001	.450	.748	.682	.834	.539	.432	.383	.174
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	[VAR_VAL_0] Initial valence variance	Correlation Coefficient	.239	.480*	.496*	.084	.379	.029	-.356	.069	1.000	.222	.140	.244	.235	-.052	-.113	.067	-.085	-.111
		Sig. (2-tailed)	.310	.032	.026	.724	.099	.905	.123	.772	.	.347	.556	.301	.319	.829	.637	.780	.721	.642
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	[VAR_TSIG_0] Initial time signature variance	Correlation Coefficient	.605**	.434	.223	-.166	.236	-.233	.282	.782**	.222	1.000	-.300	.054	.220	-.081	-.190	-.408	.018	-.387
		Sig. (2-tailed)	.005	.056	.345	.484	.316	.322	.229	<.001	.347	.	.199	.820	.351	.733	.423	.074	.939	.092
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	[VAR_MODE_0] Initial mode variance	Correlation Coefficient	-.053	.177	-.080	-.060	-.272	.059	.069	-.179	.140	-.300	1.000	-.051	.092	.021	-.001	.199	.045	.110
		Sig. (2-tailed)	.826	.454	.738	.801	.246	.806	.772	.450	.556	.199	.	.830	.699	.929	.997	.399	.851	.645
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	[VAR_KEY_0] Initial key variance	Correlation Coefficient	.006	-.242	-.077	.564**	.033	-.110	-.386	.077	.244	.054	-.051	1.000	-.186	.048	.117	.113	.005	.204
		Sig. (2-tailed)	.980	.304	.748	.010	.890	.645	.092	.748	.301	.820	.830	.	.431	.839	.623	.635	.985	.389
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	FP_5_DIFF (FP_5T - FP_5S)	Correlation Coefficient	-.094	.112	.018	-.224	.256	-.111	-.136	-.098	.235	.220	.092	-.186	1.000	-.260	-.283	-.401	-.532*	-.359
		Sig. (2-tailed)	.695	.640	.940	.343	.276	.642	.568	.682	.319	.351	.699	.431	.	.268	.227	.079	.016	.120
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	FP_6_DIFF (FP_6T - FP_6S)	Correlation Coefficient	-.141	-.171	-.097	.245	.079	-.252	.042	-.050	-.052	-.081	.021	.048	-.260	1.000	.499*	.470*	.286	.045
		Sig. (2-tailed)	.553	.470	.684	.297	.741	.285	.859	.834	.829	.733	.929	.839	.268	.	.025	.036	.221	.852
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	FR_2_DIFF (FR_2T - FR_2S)	Correlation Coefficient	-.064	-.369	-.324	.084	.088	-.322	-.020	-.146	-.113	-.190	-.001	.117	-.283	.499*	1.000	.553*	.218	.274
		Sig. (2-tailed)	.788	.109	.164	.723	.714	.166	.932	.539	.637	.423	.997	.623	.227	.025	.	.011	.356	.243
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	FR_3_DIFF (FR_3T - FR_3S)	Correlation Coefficient	-.156	.010	.048	.348	-.118	.122	-.338	-.186	.067	-.408	.199	.113	-.401	.470*	.553*	1.000	.422	.666**
		Sig. (2-tailed)	.513	.966	.841	.133	.621	.609	.145	.432	.780	.074	.399	.635	.079	.036	.011	.	.064	.001
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	FR_5_DIFF (FR_5T - FR_5S)	Correlation Coefficient	-.008	-.057	-.036	.139	-.092	-.073	.195	.206	-.085	.018	.045	.005	-.532*	.286	.218	.422	1.000	.395
		Sig. (2-tailed)	.972	.811	.881	.558	.699	.759	.410	.383	.721	.939	.851	.985	.016	.221	.356	.064	.	.084
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
	FR_8_DIFF (FR_8T - FR_8S)	Correlation Coefficient	-.179	-.250	.013	.601**	-.228	.268	-.395	-.317	-.111	-.387	.110	.204	-.359	.045	.274	.666**	.395	1.000
		Sig. (2-tailed)	.449	.289	.957	.005	.333	.254	.085	.174	.642	.092	.645	.389	.120	.852	.243	.001	.084	.
		N	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20

**. Correlation is significant at the 0.01 level (2-tailed).

*. Correlation is significant at the 0.05 level (2-tailed).

Figure E.1: Spearman's Rank Correlation Coefficient - Initial Audio Feature Variance and Relative User Experience Satisfaction

	[FILTER] VAR_INSTR_0 > 0.03	N	Mean Rank	Sum of Ranks
Initial playlist size		8	12.88	103.00
		12	8.92	107.00
[VAR_DANCE_0] Initial danceability variance		8	12.25	98.00
		12	9.33	112.00
[VAR_ACUST_0] Initial accousticness variance		8	12.50	100.00
		12	9.17	110.00
[VAR_ENERG_0] Initial energy variance		8	10.13	81.00
		12	10.75	129.00
[VAR_INSTR_0] Initial instrumentalness variance		8	4.50	36.00
		12	14.50	174.00
[VAR_LIVE_0] Initial liveness variance		8	11.88	95.00
		12	9.58	115.00
[VAR_LOUD_0] Initial loudness variance		8	9.38	75.00
		12	11.25	135.00
[VAR_SPCH_0] Initial speechiness variance		8	14.50	116.00
		12	7.83	94.00
[VAR_TEMPO_0] Initial tempo variance		8	12.88	103.00
		12	8.92	107.00
[VAR_VAL_0] Initial valence variance		8	10.38	83.00
		12	10.58	127.00
[VAR_TSIG_0] Initial time signature variance		8	12.50	100.00
		12	9.17	110.00
[VAR_MODE_0] Initial mode variance		8	11.50	92.00
		12	9.83	118.00
[VAR_KEY_0] Initial key variance		8	7.88	63.00
		12	12.25	147.00
N_ADDED_TOTAL_DIFF (N_ADDED_TOTAL_T - N_ADDED_TOTAL_S)		8	10.13	81.00
		12	10.75	129.00
ADDED_FACTOR_DIFF (ADDED_FACTOR_T - ADDED_FACTOR_S)		8	10.13	81.00
		12	10.75	129.00
N_LIKED_TOTAL_DIFF (N_LIKED_TOTAL_T - N_LIKED_TOTAL_S)		8	9.56	76.50
		12	11.13	133.50
N_REJ_TOTAL_DIFF (N_REJ_TOTAL_T - N_REJ_TOTAL_S)		8	13.19	105.50
		12	8.71	104.50
[IM1] "Music plays a significant role in my day-to-day life."		8	10.31	82.50
		12	10.63	127.50
[IM2] "My musical preferences align closely with those of the majority of music listeners."		8	14.38	115.00
		12	7.92	95.00
[IM3] "I always prefer to listen to music I have not heard before."		8	10.13	81.00
		12	10.75	129.00
[IM4] "When I find a new song I enjoy, I do not get tired of hearing it over and over again."		8	13.75	110.00
		12	8.33	100.00
[IM5] "I always know the type or genre of music I want to listen to."		8	12.25	98.00
		12	9.33	112.00
[IS1] For how long have you been a Spotify user?		8	6.25	50.00
		12	13.33	160.00
[IS2] How familiar would you say you are with the Spotify application and its functionalities?		8	10.38	83.00
		12	10.58	127.00
[IS3] On average, how many hours do you spend listening to music on Spotify?		8	13.75	110.00
		12	8.33	100.00
[IS4] How much time do you spend listening to playlists you created by yourself?		8	10.81	86.50
		12	10.29	123.50
[IS5] How often do you normally add new songs to your Spotify playlists?		8	12.25	98.00
		12	9.33	112.00
[IS6] How would you define your ideal playlist in terms of diversity of music styles or genres?		8	11.31	90.50
		12	9.96	119.50
[IS7] According to your own experience, rate Spotify's ability to offer song recommendations that fit your music taste.		8	11.63	93.00
		12	9.75	117.00
[IP1] Rate your playlist in terms of the diversity of music styles or genres in it.		8	13.25	106.00
		12	8.67	104.00
[IP2] "Listening to this playlist can often feel repetitive."		8	10.75	86.00
		12	10.33	124.00
[IP3] What percentage of this playlist was a result of Spotify's recommendation system rather than individual selection?		8	8.94	71.50
		12	11.54	138.50
[IP4] Rate the playlist in terms of your own listening experience.		8	11.38	91.00
		12	9.92	119.00
FP_4_DIFF (FP_4T - FP_4S)		8	7.56	60.50
		12	12.46	149.50
FP_5_DIFF (FP_5T - FP_5S)		8	12.56	100.50
		12	9.13	109.50
FP_6_DIFF (FP_6T - FP_6S		8	8.88	71.00
		12	11.58	139.00
FR_1_DIFF (FR_1T - FR_1S)		8	9.75	78.00
		12	11.00	132.00
FR_2_DIFF (FR_2T - FR_2S)		8	9.13	73.00
		12	11.42	137.00
FR_3_DIFF (FR_3T - FR_3S)		8	6.81	54.50
		12	12.96	155.50
FR_4_DIFF (FR_4T - FR_4S)		8	13.25	106.00
		12	8.67	104.00
FR_5_DIFF (FR_5T - FR_5S)		8	9.00	72.00
		12	11.50	138.00
FR_7_DIFF (FR_7T - FR_7S)		8	12.38	99.00
		12	9.25	111.00
FR_8_DIFF (FR_8T - FR_8S		8	5.44	43.50
		12	13.88	166.50

Figure E.2: Mann-Whitney U Test Ranks - Low/High Intrumentalness Variance Sample Split

	Mann-Whitney U	Wilcoxon W	Z	Asymp. Sig. (2-tailed)	Exact Sig. [2*(1-tailed Sig.)]
Initial playlist size	29.000	107.000	-1.466	.143	.157
[VAR_DANCE_0] Initial danceability variance	34.000	112.000	-1.080	.280	.305
[VAR_ACUST_0] Initial accousticness variance	32.000	110.000	-1.234	.217	.238
[VAR_ENERG_0] Initial energy variance	45.000	81.000	-.231	.817	.851
[VAR_INSTR_0] Initial instrumentalness variance	.000	36.000	-3.703	.000	.000
[VAR_LIVE_0] Initial liveness variance	37.000	115.000	-.849	.396	.427
[VAR_LOUD_0] Initial loudness variance	39.000	75.000	-.694	.487	.521
[VAR_SPCH_0] Initial speechiness variance	16.000	94.000	-2.469	.014	.012
[VAR_TEMPO_0] Initial tempo variance	29.000	107.000	-1.466	.143	.157
[VAR_VAL_0] Initial valence variance	47.000	83.000	-.077	.939	.970
[VAR_TSIG_0] Initial time signature variance	32.000	110.000	-1.239	.215	.238
[VAR_MODE_0] Initial mode variance	40.000	118.000	-.617	.537	.571
[VAR_KEY_0] Initial key variance	27.000	63.000	-1.620	.105	.115
N_ADDED_TOTAL_DIFF (N_ADDED_TOTAL_T - N_ADDED_TOTAL_S)	45.000	81.000	-.232	.816	.851
ADDED_FACTOR_DIFF (ADDED_FACTOR_T - ADDED_FACTOR_S)	45.000	81.000	-.232	.817	.851
N_LIKED_TOTAL_DIFF (N_LIKED_TOTAL_T - N_LIKED_TOTAL_S)	40.500	76.500	-.581	.561	.571
N_REJ_TOTAL_DIFF (N_REJ_TOTAL_T - N_REJ_TOTAL_S)	26.500	104.500	-1.673	.094	.098
[IM1] "Music plays a significant role in my day-to-day life."	46.500	82.500	-.137	.891	.910
[IM2] "My musical preferences align closely with those of the majority of music listeners."	17.000	95.000	-2.520	.012	.016
[IM3] "I always prefer to listen to music I have not heard before."	45.000	81.000	-.252	.801	.851
[IM4] "When I find a new song I enjoy, I do not get tired of hearing it over and over again."	22.000	100.000	-2.153	.031	.047
[IM5] "I always know the type or genre of music I want to listen to."	34.000	112.000	-1.134	.257	.305
[IS1] For how long have you been a Spotify user?	14.000	50.000	-2.707	.007	.007
[IS2] How familiar would you say you are with the Spotify application and its functionalities?	47.000	83.000	-.085	.932	.970
[IS3] On average, how many hours do you spend listening to music on Spotify?	22.000	100.000	-2.020	.043	.047
[IS4] How much time do you spend listening to playlists you created by yourself?	45.500	123.500	-.213	.832	.851
[IS5] How often do you normally add new songs to your Spotify playlists?	34.000	112.000	-1.173	.241	.305
[IS6] How would you define your ideal playlist in terms of diversity of music styles or genres?	41.500	119.500	-.529	.597	.624
[IS7] According to your own experience, rate Spotify's ability to offer song recommendations that fit your music taste.	39.000	117.000	-.747	.455	.521
[IP1] Rate your playlist in terms of the diversity of music styles or genres in it.	26.000	104.000	-1.721	.085	.098
[IP2] "Listening to this playlist can often feel repetitive."	46.000	124.000	-.158	.875	.910
[IP3] What percentage of this playlist was a result of Spotify's recommendation system rather than individual selection?	35.500	71.500	-.989	.323	.343
[IP4] Rate the playlist in terms of your own listening experience.	41.000	119.000	-.607	.544	.624
FP_4_DIFF (FP_4T - FP_4S)	24.500	60.500	-1.902	.057	.069
FP_5_DIFF (FP_5T - FP_5S)	31.500	109.500	-1.321	.186	.208
FP_6_DIFF (FP_6T - FP_6S)	35.000	71.000	-1.011	.312	.343
FR_1_DIFF (FR_1T - FR_1S)	42.000	78.000	-1.225	.221	.678
FR_2_DIFF (FR_2T - FR_2S)	37.000	73.000	-.882	.378	.427
FR_3_DIFF (FR_3T - FR_3S)	18.500	54.500	-2.377	.017	.020
FR_4_DIFF (FR_4T - FR_4S)	26.000	104.000	-1.729	.084	.098
FR_5_DIFF (FR_5T - FR_5S)	36.000	72.000	-.938	.348	.384
FR_7_DIFF (FR_7T - FR_7S)	33.000	111.000	-1.170	.242	.270
FR_8_DIFF (FR_8T - FR_8S)	7.500	43.500	-3.171	.002	.001

Figure E.3: Mann-Whitney U Test Statistics - Low/High Intrumentalness Variance Sample Split