# Exploration of Music Collections with Audio Embeddings

Philip Tovstogan

TESI DOCTORAL UPF / year 2022

THESIS SUPERVISORS
Dr. Xavier Serra Casals
Dr. Dmitry Bogdanov
Dept. of Information and Communication Technologies

# Acknowledgements

First and foremost, I would like to thank my supervisors: Dmitry Bogdanov and Xavier Serra. Without their experience and guidance, it would be impossible to finish this journey. I thank Xavier Serra for reminding me to keep the big picture in mind and be able to see the forest behind the trees. I thank Dmitry Bogdanov for his expertise, teaching and guiding me in the ways of auto-tagging, music recommendations, and not forgetting to treat music as music and not just generic data. He helped me the most throughout my Ph. D. to do rigorous research and not be afraid to investigate less popular topics. He was always supporting me and helping me to figure out the next steps whenever I was stuck. I also thank Perfecto Herrera for his expertise and for helping me design user studies ethically and systematically.

During the COVID-19 pandemic, I have been primarily working remotely. I am still thankful for sharing the office with Antonio Ramires and Andres Ferraro when we were still working from the office. That was the most pleasant office to work in with our discussions, both work-related and small talk, making tea, and going for the coffee breaks.

I thank all my colleagues from MTG for fruitful discussions and extraordinary times that we spent together in the university and outside (in no particular order): Alastair Porter, Frederic Font, Xavier Favory, Pablo Alonso, Pablo Zinemanas, Eduardo Fonseca, Minz Won, Furkan Yesiler, Jordi Pons, Marius Miron, Vsevolod Eremenko, Jyoti Narang, Albin Correya, Olga Slizovskaia, Alia Morsi, Benno Weck, Genis Plaja, Guillem Cortes, Jorge Marcos Fernandes, Juan Gomez, Lorenzo Porcaro, Luis Joglar-Ongay, Miguel Perez, Roser Battle Roca, Thomas Nuttall, Xavier Lizarraga, Sergio Oramas.

I am also thankful to the administrative personnel of MTG and UPF that helped me with way too many questions and bureaucratic processes: Cristina Garrido, Sonia Espi, Lydia Garcia, Montse Brillas, Lluis Bosch, and Marcel Xandri.

I can't express how thankful I am to all organizers of the MIP-Frontiers program European training network that provided funding for this Ph.

D.: Xavier Serra, Gaël Richard, Emmanouil Benetos, Geoffroy Peeters, Simon Dixon, Gerhard Widmer, Sven Ahlbäck, Stefan Lattne. MIP-Frontiers provided so much more than just a Ph. D. program with numerous training in hard and soft skills. I am immensely thankful to Alvaro Bort, the coordinator. He was always in contact with everybody regarding all the aspects of the project and provided answers to all administrative questions that I had. And of course, to all my fellow peers: Karim M. Ibrahim, Kilian Schulze-Forster, Giorgia Cantisani, Ondřej Cífka, Javier Nistal, Emir Demirel, Carlos Lordelo, Ruchit Agrawal, Alejandro Delgado, Vinod Subramanian, Luís Carvalho and Charles Brazier.

The following line has been with me throughout the project and will forever serve as a thread that connects everybody that was participating in it:

> This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 765068 (MIP-Frontiers).

Last but not least, I wish to thank my partner Claudia Escalona for always being by my side. Her love and support kept me going through all steps of this journey, especially in the darkest moments. I am very thankful to my father Yuri Tovstogan and my mother Tetiana Schchebetova for everything. And, of course, I thank my friends for their continuous support and my cat Pancito for his emotional support.

# Abstract

Music recommendation systems (RecSys) are integral to modern music streaming services. While there is much research on many aspects of RecSys, there is not enough research on exploration and discovery that contributes to long-term user retention. After conducting an anonymous survey, we identify that the exploration and rediscovery of the personal collections in particular needs improvement. To address this, we take advantage of music tags (genre, moods) and use deep auto-tagging systems to construct latent spaces. We investigate different architectures, datasets, layers, and projections and how they affect the perceived similarity of nearest neighbors. Finally, we present a novel web interface to visualize music collections using audio embeddings. We evaluate the proposed solution via semi-structured user interviews and conclude that it provides an excellent alternative to existing solutions. We believe that the contributions of this work enable more research and industry solutions for music exploration and discovery.

# Resum

Els sistemes de recomanació de música (RecSys) son una part integral de les actuals plataformes de música en streaming. Tot i que s'ha fet investigació sobre molts aspectes relacionats amb RecSys, encara falta investigació sobre l'exploració i el descobriment de continguts que permeti fidelitzar usuaris a llarg plaç. Després de realitzar un estudi preliminar, hem vist que existeix una manca d'eines per al re-descobriment de les col·leccions de música personals. Per abordar aquest problema, en aquesta tesi ens focalitzem en l'us d'etiquetes musicals sobre estil i mood i treballem en espais latents de dades entrenant predictors automàtics d'etiquetes basats en models d'aprenentatge profund (deep auto-tagging systems). Analitzem i comparem diferents arquitectures de xarxes neuronals, bases de dades, i diferents tècniques de projecció de dades per entendre com aquestes afecten al concepte de similaritat percebuda entre peces musicals que han estat projectades en punts propers dels espais latents. Finalment, mostrem una interfície web que hem desenvolupat per visualitzar i navegar col·leccions de música utilitzant els espais latents. Hem avaluat aquesta interfície a partir d'entrevistes semi estructurades i hem conclòs que la interfície proporciona una alternativa excel·lent als sistemes tradicionals de navegació de col·leccions musicals. Creiem que les contribucions d'aquesta tesi permeten que es desenvolupi més recerca i es creïn més aplicacions industrials per abordar el problema de l'exploració i descobriment de música.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# INTRODUCTION

Music is a massive part of the human culture throughout the world. Everybody knows what music is and the majority of people listen to music on regular basis. Even if there are people who don't listen to music, they are still exposed to it because of the culture and society. The way people listen to music evolved a great deal throughout human history: from in-person live performances to the digitized collections of the most of the recorded music available at our fingertips.

Contemporary music streaming services are the primary source of music consumption and discovery (IFPI, 2021). And almost every music streaming service company has an embedded recommendation system to suggest the music to listen that is usually the primary way for users to discover new music. The goal of the most of the recommendation systems is to suggest the tracks that are predicted to be liked by the user the most based on various data available. However, it is also possible for user to discover and engage in the completely new types of music that wouldn't be related to their tastes, which is difficult to predict for typical recommendation systems.

The process of recommendations exhibits the *explore-exploit dilemma* (Barraza-Urbina, 2017). This dilemma is widely known from multi-armed bandit problem from game theory and reinforcement learning (Auer et al., 2002). *Exploitation* involves recommending tracks which adhere to user's

tastes with high confidence, thus have high probability to be liked. This is a safe approach with short-term reward, thus it contributes greatly to user retention. For the most of the music streaming companies the exploitation behavior of recommendation systems aligns well with the business model. Because the goal of exploitation is to predict tracks that the user will like, repeated recommendations are a common occurrence.

An opposite behavior to this, the goal of music *exploration* is to provide the user with novel and horizon-broadening tracks (even if not necessary liked) with a potential for the user to find completely new favorite genres or artists. Exploration can potentially have a long-term reward in introducing the user to a new genre or type of music, but it carries more risk with it. In the industry, streaming companies spend a lot of resources on improving the exploitation and to find an acceptable balance between exploration and exploitation (Barraza-Urbina, 2017) that is visualized on Figure 1.1. Having an exploration algorithm that exhibits better performance than random baseline takes a lot of resources, thus it is much more difficult for the industry to warrant spending resources on improving exploration algorithms.

Thus, to summarize, exploitation usually provides benefits in short term, while exploration has a potential to contribute in long term. In addition, exploration is a more active experience, as it requires effort and attention by the nature. On the contrary, exploitation is usually associated with passive experiences, as it takes advantage of familiarity. In current streaming platforms there are multiple sources of exploitation, while the exploration is usually more difficult to access.

Moreover, it is easier to evaluate the success of the exploitation than of exploration. Typical offline recommendation systems metrics (accuracy, precision, recall, NDCG) measure the performance of exploitation. Although in industry it is common to use online metrics such as click-through-rate, time spent on platform, etc. (which can capture exploration performance), those are much less present in published academic literature. Comparatively, there is less research done on the exploration than on exploitation, thus we want to focus on *exploration*. Other terms that are often used include discovery, browsing, navigation, etc. There are nu-

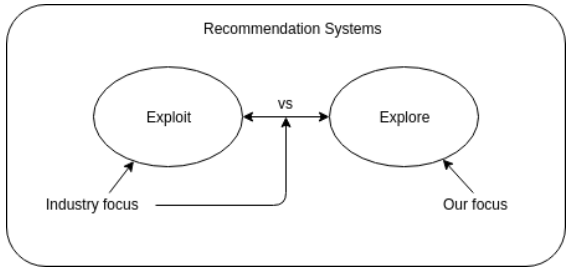



Figure 1.1: Exploit vs Explore

ances to each term, but the essential concept is to find new music that is far away from the obvious preferences. While it is certainly possible to explore music within the scope of individual tastes (e.g. digging), it is much more active and involved process.

The discovery of a completely new music that you like is a magnificent experience. Thus, in this thesis we want to take a closer look at methods for music exploration and discovery and to address some problems with the way it is currently treated in music streaming services as well as outside.

## 1.1 Recommendation systems

The goal of recommendation (or recommender) systems (RecSys) is to suggest new items to users. Usually, they infer the preferences based on the history of interaction with other items. The research in the field of recommendation systems has been steadily rising since 2001, although there was an early work by Shardanand (1994). In the industry, with the adoption of recommendation systems into e-commerce (Amazon, Netflix) leading to higher user retention it became hot topic very quickly. There are numerous approaches to recommendation (Ricci et al., 2022), but the two fundamental are: collaborative filtering and content-based recommendations (Brusilovsky et al., 2007).

*Collaborative filtering* (CF) in general can be described as a technique of filtering information that involves collaboration between multi-

ple agents, data sources etc. The basic assumption of CF is that if user A has similar opinions to user B, then it is more likely that user B has opinion of user A than of random user. Thus, in regards to music recommendation CF-based approach can use the target user's preferences to identify other users that have similar preferences to target user, and then recommend the missing tracks that other users have in their preferences, but the target user doesn't. It was first successfully used for email sorting system in Goldberg et al. (1992), first as a music recommendation algorithm in Cohen and Fan (2000). Collaborative filtering works very well even without considering the domain knowledge. The state-of-the-art CF approach now comprises of various matrix factorization (Koren et al., 2009) algorithms that operate on user-item matrices of interaction statistics. There are many deep learning based approaches that are proposed that are achieving state-of-the-art (Martins et al., 2020).

However, several shortcomings of collaborative filtering include *cold-start problem* (Maltz and Ehrlich, 1995), where if tracks don't have any listening data, they will not be recommended to anyone and thus have difficulties gaining more listening data. So this cycle is difficult to break unless the tracks explicitly will gain a minimal amount of listening data to be recommended. Closely related to this problem is *popularity bias* that makes it difficult for less popular music to be recommended. And another problem, called *filter bubble* (Pariser, 2011) is caused by system essentially trying to make all users with slightly similar taste to listen to similar music, i.e. being in the bubble, and it becomes difficult for users to escape the bubble, thus exacerbating overall segregation and clustering in the userbase.

While CF leverages user data to do recommendations, *content-based* (CB) recommendations use the information available about or extracted from the content itself. One of the first CB systems was used to recommend technical reports based on previous technical reports read in Bellcore (Foltz and Dumais, 1992). At the same time one of the first music recommendation system — LyricTime was using lyrics to learn a user profile and recommend music based on it (Loeb, 1992). Content for CB can be metadata, categories, text or in case of music - features extracted

with music information retrieval algorithms. CB does not exhibit the cold-start problem, moreover, it is a solution to it. However, CB is more expensive from the computational standpoint what limits its scalability. It is safe to say that modern recommendation systems are *hybrid*, mostly using CF as primary recommendation engine and using CB to deal with the cold-start problem and other shortcomings of CF (Yoshii et al., 2006; Liang et al., 2015; Wang and Wang, 2014).

Talking about *novelty and exploration* aspect of recommendation systems, Oramas et al. (2017b) had done some work on the recommendation systems with some evaluation metrics based on the novelty (Bellogín et al., 2010), specifically for the evaluation of exploration systems. Celma and Herrera (2008) had shown how popularity bias of CF systems can hinder the novelty of the recommendations, but while CB doesn't exhibit it, perceived quality of the recommendations by users was higher for CF system. Schedl and Hauger (2015) extensively explored notions of diversity and novelty as an important factors in music recommendations. Ferraro et al. (2021c) address the fairness in the music RecSys, particularly from the artists' perspective. Overall, it is important to use user-centric evaluations (Schedl and Flexer, 2012) in this kind of research because it doesn't matter if objective novelty metric is high if it hurts user experience and enjoyment.

Li et al. (2019) researched user search behavior in the context of music streaming services and identified two mindsets: *focused* and *non-focused*. In the focused mindset, users know what they are looking for; and in non-focused, they only have a rough idea. While it was studied in the context of the complete catalog of the music available on the streaming services, those mindsets also apply to the case of users that mostly listen to their personal collection. One of the situations that we want to consider is when the user doesn't know what he wants to listen to (non-focused) but wants to listen to something familiar (from personal collection) — *rediscovery*. The topic of rediscovery is usually not separated from discovery in streaming platforms, i.e. user can get tracks from their library in the "discovery" playlists. However, in this thesis we want to pay close attention to the following situation: *The user doesn't know what he wants to*

*listen to (non-focused mindset) but wants to listen to something familiar (from personal collection).*

CF is much more dominant approach for implementation of RecSys in online platforms, as it captures the exploitation behavior very well. We mostly focus on CB methods in this thesis, as it is more suited for the exploration behavior and doesn't require history of user interactions that results in a platform-agnostic approach.

## 1.2 Personal music collections

As streaming have grown to be the primary way of music consumption with access to massive library for a recurring payment, the concept of personal music collections has been slowly disappearing (Cunningham and Cunningham, 2019). Before, it was quite common to have most of the music collection as CD albums. With the appearance of iPods and portable MP3 players, the collections of the CDs were transformed and transitioned into digital library of MP3s that could be taken anywhere. With the rise of smartphones, the digital collections slowly moved from MP3 players towards the phones, but they still remained collections that were gathered by the users.

Now, the paradigm of music discovery in streaming services neglects the listeners who might want to re-engage with their personal music collections, gathered, curated, and appreciated by their maintainers throughout the years (Cunningham and Cunningham, 2019). For such users, exploring their own curated music selections can be a pleasurable and rewarding experience, helping to appreciate and re-contextualize relations between music items and rediscover artists or tracks that they haven't listened to in a long time.

It can be especially relevant in the context of digital music downloads, which still have a considerable impact within independent music distribution (IFPI, 2021) (e.g., Bandcamp[1] has gained growing digital sales over the past years with a strong following among music enthusiasts).

[1] `bandcamp.com`

In this context, many music consumers, and also musicians, DJs, radio hosts, music journalists, archivists, and other professionals or hobbyists that work with digital music collections can benefit from exploration and rediscovery functionality.

For the users that like to have their own media servers, there are many solutions to stream from their server to devices (e.g. Subsonic[2], Plex[3]). There is a community of people supporting this independent direction too, with people building communities of music metadata, such as MusicBrainz.[4]

Cunningham and Cunningham (2019) argue that the disappearance of the concept of the personal music collection devalues music moving it to utilitarian commodity. If before people spent time and effort in maintaining their physical collections, now it is difficult to warrant the effort required for even the people that have time and desire to do that, given the convenience of listening to music on streaming services. The shift from personal collection to playlists, following and liking artists and tracks is encouraged on the streaming services by design. In this thesis, we want to keep the notion of personal music collections, whether it relates to actual owned music, or albums and tracks from the streaming services that are "added to the personal library".

## 1.3 Music discovery

### 1.3.1 Streaming platforms

The interfaces for music exploration and discovery are quite homogeneous in the industry. The recommendations are usually presented in the form of the playlists or artists, and if the user wants to browse their personal collections, you are presented with options to see your playlists, artists or albums. Once you go to artist, the concept of personal collection

[2] `http://subsonic.org`
[3] `plex.tv`
[4] `musicbrainz.org`

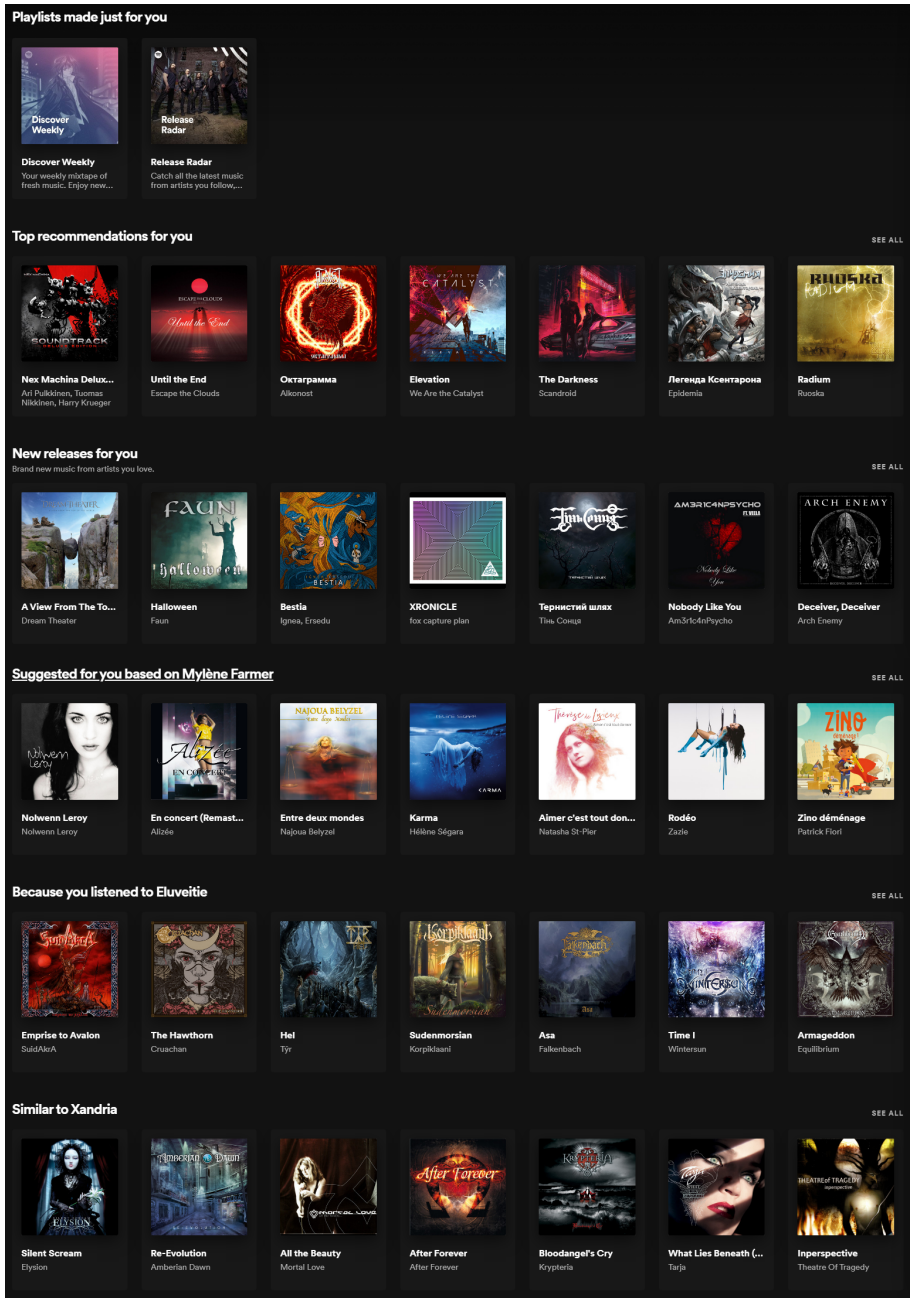



Figure 1.2: “Discover” section on Spotify

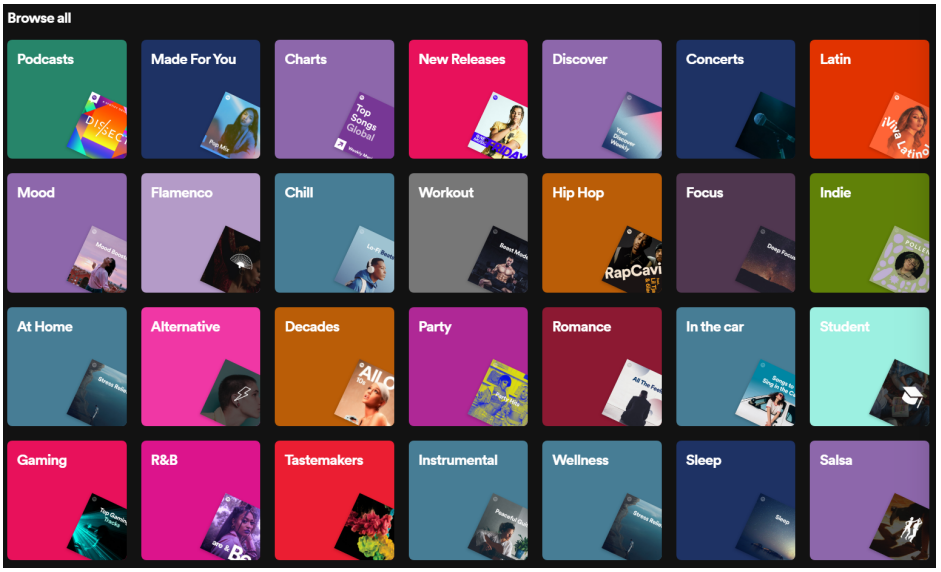



Figure 1.3: “Browse all” section on Spotify

is already almost non-existent, as you are presented with all works of the artist. While you can usually see the songs liked by you of this artist, they usually comprise a playlist. The groupings of artists by genre are replaced by genre playlists, which also immediately takes you away from personal collections.

Now that it is evident that the concept of personal collections is discouraged in the streaming platforms, what is the intended way to find some music to listen to? One way is the discovery from exploitation that is achieved by recommender systems, and is usually provided to user in form of playlists, artists or albums: “Discover Weekly”, “Top recommendations to you”, “Suggested to you because you listened to . . . ”, “Because you listened to . . . ”, “Similar to . . . ”, “More like . . . ” (see Figure 1.2).

However, if the user doesn’t want anything that is similar or based on music that they usually listen to, the only option is to browse the full catalog (see Figure 1.3). The categories that can be useful for browsing are *genres*: latin, flamenco, hip-hop, indie, alternative, R&B, salsa;

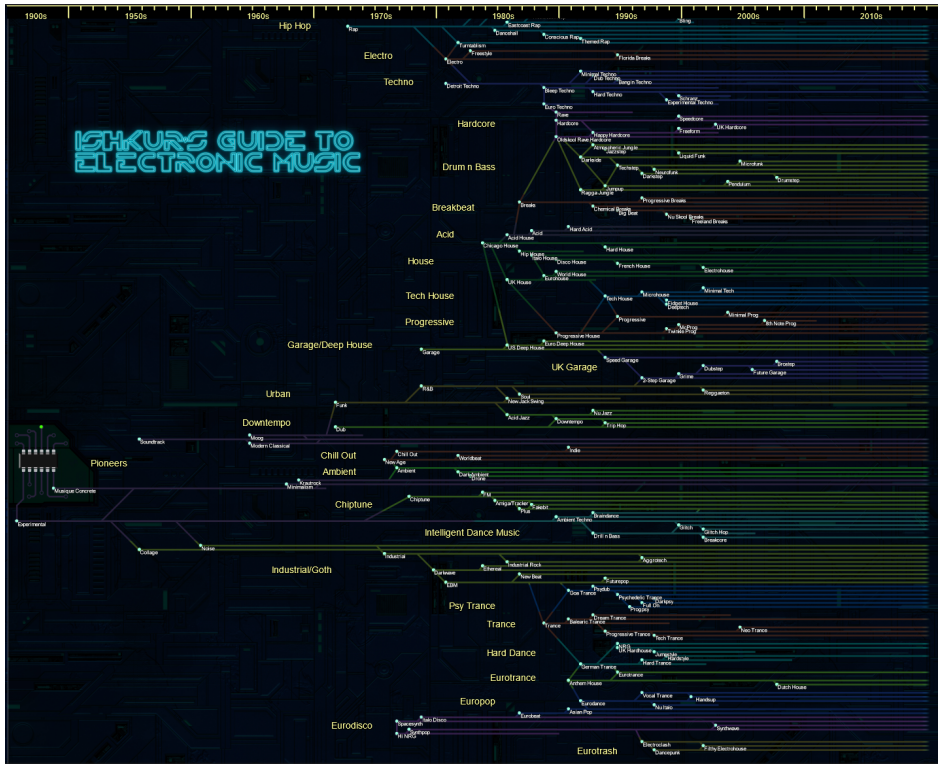



Figure 1.4: Ishkur's Guide to Electronic Music

*moods*: mood, chill, focus, party; *contexts*: at home, workout; etc. All these categories and tags have many playlists that can be used to explore and discover new categories. These and many more tags are also used to categorize music.

### 1.3.2 On the web

Outside of the streaming services, there are many ways to explore music online. Some obvious places for that are music databases and review

websites like Wikipedia, Discogs[5], MusicBrainz, RateYourMusic[6], AllMusic[7] etc. Regular social networks have a lot of music communities to explore and Last.fm[8] that was created in 2002 as a music database is now a social network for music, where you can scrobble your listens and see a lot of statistics about your listening habits. Online music journalism is very prominent and the well-written articles can serve as a starting points to dive deeper into genres or artists, and they usually include embedded track previews.

One of the first websites specifically for music exploration is *Ishkur's Guide to Electronic Music*[9] created in 1999 as a Flash website. It presented the genealogy of electronic music throughout the years including 153 genres and 818 audio files visualized with connections between genres and a timeline (Figure 1.4) that can be navigated and listened to.

Similarly, *MusicMap*[10] provides a genealogy of popular music genres including the relations, timeline, and evolution. It is a great visual tool to explore new genres as well as the genres that you like and find the ones that are related and potentially discover new music (Figure 1.5). It is very comprehensive, but it suffers from popularity bias as well as being biased to western music. Music that is used as examples for different genres is curated and can be considered a good representation of that particular genre.

Another genre exploration tool is *Every Noise at Once*[11] that maps all possible genres onto 2D plane with similar ones grouped together. By clicking one genre user can hear a sample, and it is also possible to expand the genre that caught interest and see the list of artists. While the neighborhoods of familiar genres will probably be familiar, the website has a functionality that will randomly pick one genre and play the sample, and keep iterating.

[5] `discogs.com`
[6] `rateyourmusic.com`
[7] `allmusic.com`
[8] `last.fm`
[9] `music.ishkur.com`
[10] `musicmap.info`
[11] `everynoise.com`

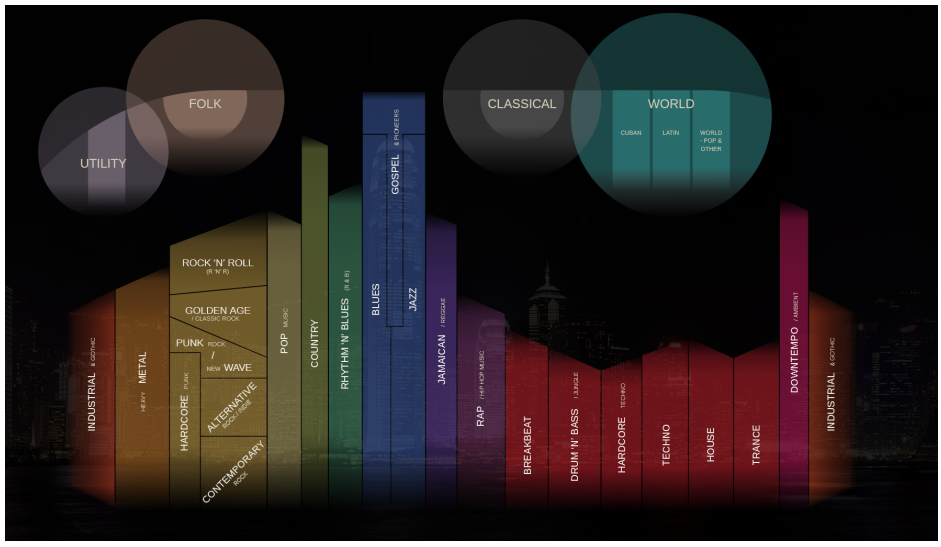



Figure 1.5: MusicMap

Figure 1.6: Every Noise at Once (screenshot of a sample)

Above-mentioned exploration tools are useful for generic music exploration, *Discover Quickly*[12] facilitates the rediscovery of the personal music collection by playing audio on hover of the cover art. Not only users can rediscover their personal collections, this tool is also useful for browsing genres and playlists.

## 1.4 Auto-tagging

Music auto-tagging is the process of assigning various tags to a piece of music. Music auto-tagging is the multi-label classification task that usually encompasses multiple categories of tags. However there are many tasks that are single-label classification problems: genre classification, mood classification, etc. (Fu et al., 2011). Auto-tagging takes audio as input and generates tags for the music: genres, moods, themes, etc.

One of first papers in this research area is by Tzanetakis (2001) in genre classification. It tackles the problem as a single-label classification task with 5 genre labels: *classical*, *country*, *disco*, *hip-hop*, *jazz* and *rock*. Since 2001 and with the rise of the deep learning, auto-tagging systems became more prominent and hot topic in music information retrieval (Nam et al., 2019).

Not only the tags are commonly used for music discovery and exploration, the auto-tagging systems can also be used for music recommendation systems, particularly for CB approaches. The playlists in music streaming systems often are based on one or more category of tags, for example genres, moods, decades, contexts. Some examples of multiple tags being used: *metal workout*, *80s rock*, *relaxing piano*.

As tags are quite useful for music exploration and discovery, in this thesis we investigate how deep auto-tagging architectures can facilitate the process of music exploration and rediscovery.

[12] `discoverquickly.com/`

## 1.5 Survey

To learn more about the people's music listening, exploration and discovery habits, we conduct the anonymous survey. The questions are listed in Appendix B and the survey is built using Google Forms. We made sure that there is no personal information is being gathered, and explicitly asked participants to not provide any personal identifiable information, and manually monitored the free-form answers to delete any responses that had any (2).

We circulated the survey in the social media (Twitter, Reddit) and relevant mailing lists and in total received 319 responses. 50% of respondents identify as men, 38% as woman, 2% preferring not to say and 10% as other. 38% are aged 18–24, 39% — 25–34, 15% — 35–44, 5% — 45–54, 3% — 55–65, and less than 1% preferring not to say. The respondents of the survey are mostly based in western countries (36% USA, 10% UK, 9% Spain, 7% Canada, 6% Germany, 4% France, etc.), and there are in total 43 unique countries.

While respondents generally agree that there are a lot of options for music exploration and discovery, the amount of respondents that are satisfied with the existing options is noticeably less (see Figure 1.7). Many respondents have answered that is not easy for them to get an overview and manage their library, and similarly other questions about interaction and rediscovery have higher amount of negative responses.

Figure 1.8 shows the distribution of answers to the following three questions:

- How often do you have a desire to listen to new music?
- How often would you like to listen to something from your collection/library that you haven't listened in a long time?
- How often do you ACTUALLY listen to something from your collection/library that you haven't listened in a long time?

We can see that people's desire to discover new music is slightly more frequent than to rediscover their music, although not that much differ-

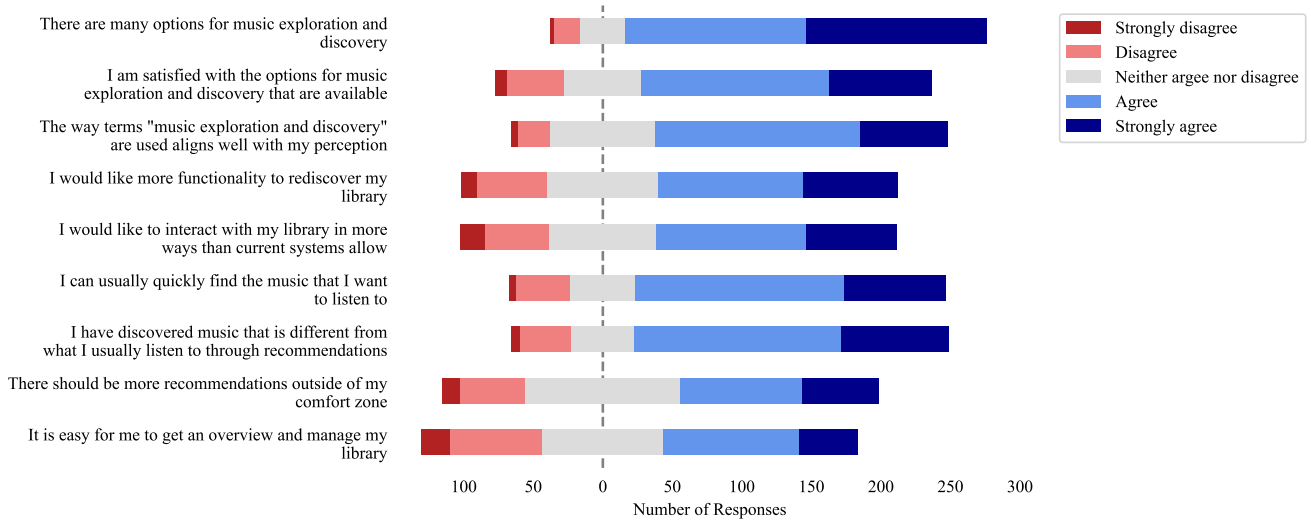



Figure 1.7: Statements about music streaming services

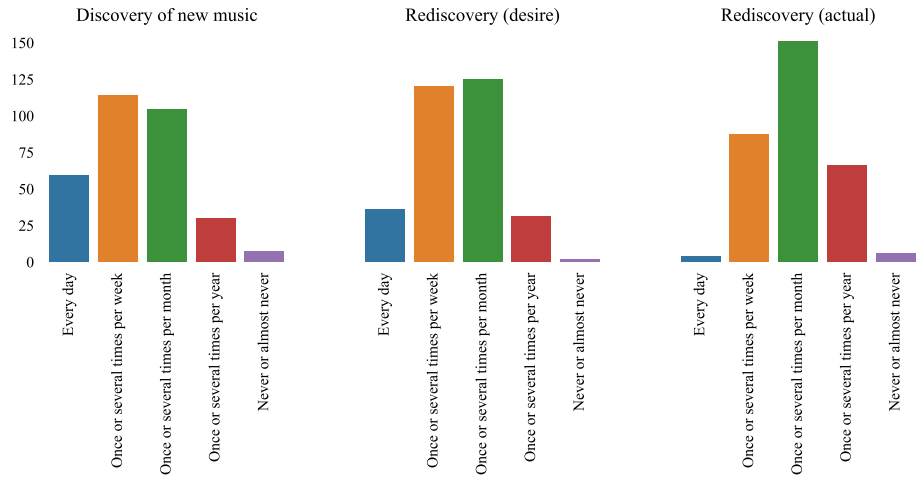



Figure 1.8: Frequency of discovery and rediscovery

ent. Moreover, we can see that people don't always act on their desire to rediscover, as there is a shift towards less frequent options. Out of 127 respondents (total of 319) that engage in rediscovery not as often as they have desire, the following reasons have been commonly mentioned:

- Not being in the proper mood (28)
- Forgetting (28)
- Prioritization of new music, enjoying discovery more than rediscovery (25)
- Being lazy, going for easy options from home page, the desire being fleeting, not caring too much, requiring effort, being too used to familiar or regular music, passivity, etc. (25)
- Shuffle algorithm prioritizing the recent music (15)
- Not having enough time (14)

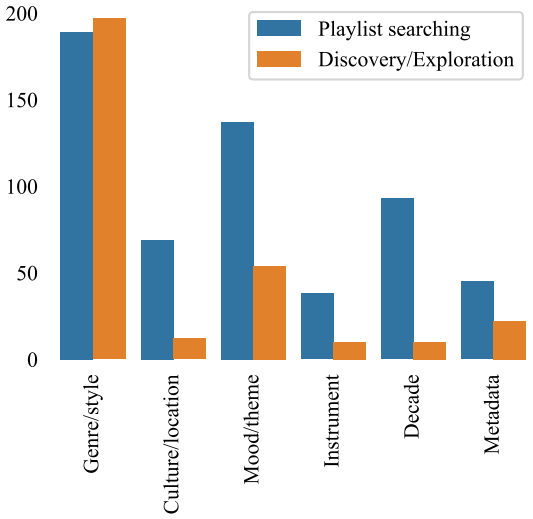



Figure 1.9: Terms used for playlist search and exploration/discovery

- Listening too much to that music, overplaying it, getting bored, burned out or having preferences and tastes change (13)
- Navigation of the personal collection being not good for rediscovery (13)
- Library being too large (5)
- Difficulty of accessing physical media (3)

We can see that among the four most commonly mentioned reasons for not acting on the rediscovery desire is not being in appropriate mood. We can also see that multiple respondents (15) have mentioned that they notice that shuffle algorithms are prioritizing recently played music, and they don't like that, as it makes rediscovery a process that you need to actively engage in, as opposed to more passive shuffle.

Figure 1.9 shows the different terms that users use for playlist search and exploration. For the question of playlist search, multiple choices

could be picked, while for exploration/discovery question only one answer could be chosen. However, we can see the relative importance of the terms, with genre/style being the most used one, and mood/theme being the second. The relative difference between genre and mood is much greater for discovery/exploration compared to playlist searching. That is interesting, as we can draw conclusion that moods are less important then genres for exploration.

When asked about sources for discovery (participants could choose more than one option), out of 319 respondents 76% (244) use the streaming platform discover functionality, 66% (211) — social recommendations, 28% (95) — music identification apps, 26% (84) — music journalism, 10% (36) — influencers and journalists, and 9% (30) – music stores. Among the "other" answers provided by the respondents commonly mentioned are radio (12), background music and recommendations from social media like YouTube, Instagram, TikTok (13), Reddit and other forums (12), hearing music in TV shows, movies and videogames (7), using third-party tools specifically for music discovery (e.g. Boil the Frog, Every Noise at Once, Rate your Music) (7).

Talking about the strategies for music discovery, the respondents are spread quite evenly between the provided options: 31% know what they are looking for, 35% have a vague idea, and 25% have no idea. A lot of free-form responses are either all of the above, or some combination of the the options, depending on the context. There are few responses that indicate that they like to just let the recommendation systems do their work.

To summarize, there are indeed many options to explore and discover music. There is a spectrum of people from the ones that like to let the recommendation systems do all the work for them in terms of discovery, towards the ones that put a lot of effort into discovering new music. Moreover, talking about the rediscovery, the streaming services do not encourage this behavior, while certain population of people clearly has the desire for it.

## 1.6 Problem statement and thesis organization

While there are many ways to facilitate music exploration and discovery, as we see from the survey, genre and mood tags are one of the most used ones. As tags are ubiquitous in their ability to capture many facets of music, we focus on using auto-tagging systems as the technology to facilitate and enable music exploration and discovery. We have two main research question we want to address:

- RQ1: can auto-tagging systems learn music representations that will be useful for music exploration and rediscovery
- RQ2: can these representations help visualize the music in the way that can be beneficial for user to explore or rediscover music

We start with introducing new auto-tagging dataset in Chapter 2. In Chapter 3 we explore the modern auto-tagging deep learning architectures and summarize three years of organizing a task in MediaEval challenge. Chapter 4 talks more in depth about similarity spaces. Then we introduce and evaluate new proposed music exploration interface in Chapter 5. Chapter 6 summarizes the work and contributions of this thesis, as well as discusses the implications and possible future directions.

# Chapter 2

# AUTO-TAGGING DATASETS

## 2.1 Introduction

To train good music auto-tagging models in a supervised fashion large amount of annotated data is required. It is not a problem in the industry because of large catalogs of commercial music with metadata and annotations available. However, it is not easy to share commercial music for research purposes due to copyright regulations. There are several open datasets that are trying to address this issue in different ways: including audio features instead of audio or providing audio upon request.

One example of the platform providing audio features is AcousticBrainz[1] (Porter et al., 2015). This platform allows users to extract audio features for their personal music collections and upload them. The features vary from low-level ones such as MFCCs, HPCP, loudness, etc., that are represented through their statistics over the length of the track (mean, variance, kurtosis, etc.) to high-level ones such as key, tempo, danceability, genre, etc. However, the limitation is that the features are not provided on a per-frame basis but only as an overall summary. With the rise of deep learning and the amount of data required to train deep models, it is not enough to get good performance.

[1] `acousticbrainz.org`

| Name | Tracks | Artists | Tags | Audio | Split |
|---|---|---|---|---|---|
| Million Song Dataset | 505 216 | - | 522 366 | N/A* | 1* |
| MagnaTagATune | 25 877* | 230 | 188 | Poor | No |
| Free Music Archive | 106 574 | 16 341 | 161 | Good | 1 |
| Music4All | 109 269 | 16 269 | 19 541 | Good | No |
| Melon Playlist Dataset | 649 091 | - | 30 652 | Spec. | No |
| **MTG-Jamendo** | 55 609 | 3 565 | 195 | Good | 5 |

Table 2.1: Auto-tagging datasets

The following section introduces the most important and widely used open datasets for music auto-tagging and their typical use cases and known limitations.

## 2.2 Existing datasets

It is not easy to talk about auto-tagging without mentioning *GTZAN* (Tzanetakis, 2001), which is one of the first music classification datasets. While now primarily used in the demos and tutorials, it was one of the first labeled audio datasets. It contains $15 \times 50 = 750$ tracks that span 15 genres with 50 tracks per genre. Every track is 30 seconds, so in total, there are $750 \times 0.5 = 325$ minutes of audio. While it is perfectly balanced and relatively small, there are many issues with this dataset (Sturm, 2014): repetitions, mislabeling, and distortions.

*Million Song Dataset*[2] (MSD) (Bertin-Mahieux et al., 2011) is one of the most famous auto-tagging datasets due to it's size. However, one of its biggest issues is that there is no audio publicly available. When it was released, it was possible to download 30 seconds of audio previews with the 7digital service, which is not available anymore. The previews ranged in quality with an average of 104kbps bitrate and a sampling rate of 22 or

[2] `http://millionsongdataset.com`

44.1kHz.

The tags are taken from Last.fm[3] and are generated by the users. Given the number of tracks, tags and that source of the tags are users of Last.fm - the tags are quite noisy (Choi et al., 2018). With the sheer number of tags present in the full dataset, it is not uncommon for researchers to use the top 50 tags from the MSD dataset. One of the notable splits was created by Choi et al. (2016). The split[4] contains 242 854 tracks: 201 680 training, 12 634 validation and 28 540 testing. The obvious tags that are not related to music content have been discarded. The valid tags include genres (rock, pop, jazz, funk), eras (60s – 00s), and moods (sad, happy, chill). More recently, Won et al. (2021) have identified and addressed issues with previous split, and released new cleaned and artist-level stratified split (CALS)[5]. It contains 233 000 labeled tracks and 516 000 unlabeled tracks, enabling it to be used in semi-supervised learning. The labeled 233 195 tracks are separated into 163 550 training, 34 730 validation, and 34 915 testing sets.

*MagnaTagATune* (MTAT) (Law et al., 2009) is a smaller dataset that is typically used for prototyping of the auto-tagging systems. The authors used a game approach to ask two users to tag the track and then answer if they think they had the same track to tag. The resulting annotations include 188 tags. The website provides audio to be downloaded as well as metadata and annotations. The provided audio does not include full tracks but the 25 877 segments that have been annotated. While having audio available is an advantage, the audio quality is quite low: 32kbps and mono.

Some issues with MagnaTagATune is that if you use it out of the box, there are multiple groups of redundant tags. For example, even in top 50 tags: *vocal*, *vocals*; *no vocals*, *no vocal*, *no voice*; *female vocal*, *female voice*, *woman*; *man*, *male*, *male voice*. This stems from allowing free-form annotations by the users instead of using standardized vocabulary. Similarly to MSD, given the sparsity of the tags, researches often use top

[3] `last.fm`

[4] `github.com/keunwoochoi/MSD_split_for_tagging`

[5] `github.com/minzwon/semi-supervised-music-tagging-transformer`

50 tags.

*Free Music Archive* (FMA) (Defferrard et al., 2017) was introduced to address the lack of large datasets with high-quality audio. It contains 106 574 tracks from the internet music archive licensed under the Creative Commons (CC) license, thus having no problems providing the audio of full tracks. While the tags only include genres and sub-genres, the audio is high-quality: up to 320kbps, 263kbps on average, 44.1 kHz sample rate, and stereo.

The issues with FMA are the lack of curation of the collection, as there is much music of questionable origin and noisy tags. Many recordings are of low technical quality and not up to the current industry standard of mastering and quality control in music distribution.

Several large-scale auto-tagging datasets were introduced later. One of such is *Music4All* (Pegoraro Santana et al., 2020) that has been created by scraping Last.fm for users, tracks and tags, and YouTube for audio. The dataset contains 109 269 tracks annotated with 19 541 tags, 853 of which are genres. It has been linked to Spotify tracks ids and contains the audio features that are available from Spotify[6] API (such as valence, energy, liveliness, etc.). The dataset contains audio that is middle 30 seconds of the track. However, to get the dataset, one needs to request it from the authors.

Another dataset worth mentioning is *Melon Playlist Dataset* (Ferraro et al., 2021b) that has been created for the task of automatic playlist continuation in collaboration with the Korean streaming service Kakao. The dataset contains 649 091 tracks from 148 826 playlists. The tracks are annotated with 30 652 tags, with the number of unique genres being 30 and sub-genres — 219. The dataset distributes mel-spectrograms extracted from each track's 20-50 seconds of audio.

[6] `spotify.com`

## 2.3 MTG-Jamendo dataset

To address the common issues with open music auto-tagging datasets, we introduce the new *MTG-Jamendo* dataset. It includes full tracks from Jamendo Music[7] that are licensed under Creative Commons in high quality. The tags are provided by artists and curated by Jamendo. There are multiple categories of tags: genres, moods/themes, and instruments. Also, as all tracks are present on the Jamendo platform to be listened to by users or used in a commercial application, the basic level of mastering quality and absence of artifacts is ensured, so the quality of music is closer to the commercial music collections.

We started with 56 639 tracks and filtered out tracks that were less than 30s in duration, resulting in 55 701 tracks. We then encoded them in high-quality 320kbps MP3, resulting in 509 GB of audio. The median duration of a track is 224s, and in total, there are 3 777 hours of audio.

All these tracks are annotated by in total of 692 tags. As some tags have the same meaning, we wanted to merge some tags. However, the question was where to draw the line on merging. Obviously we can merge *relax* and *relaxing*, but should we merge *electronic* (electronic music in general) and *electronica* (specific sub-genre of electronic music)? Thus we employed three researchers to look through the list of tags separated by the category and independently give an opinion on which tags could be merged. Then we merged only the tags that all three researchers agreed on. The exact mapping can be found in the GitHub repository. Essentially, we merged tags to consolidate variant spellings, translations, and tags with the same meaning, re-mapping 99 tags (less than 15%). Examples include *synth* to *synthesizer*, *guitarra* to *guitar*, *soundtracks* to *soundtrack*. After the tag merging stage, the number of tags decreased from 692 to 595.

Another issue with tags was that there are some tags that are too specific and only used once or several time, for example *deutschrock*, *lyre*, *surdo*, *guilty*, *awake* that were used only by one artist per tag. For auto-tagging, these tags are tough to learn, as there are not represented by

[7] `jamendo.com`

| Category | Tags | Tracks | Albums | Artists |
|---|---|---|---|---|
| Genre | 87 | 55 094 | 11 186 | 3 546 |
| Instrument | 40 | 24 976 | 5 672 | 2 003 |
| Mood/theme | 56 | 17 982 | 4 423 | 1 508 |
| All | 183 | 55 525 | 11 256 | 3 565 |
| Top-50 | 50 | 54 380 | 11 107 | 3 517 |

Table 2.2: Statistics for category subsets

enough data. Thus, we filtered out all tags with less than 50 unique artists represented and kept the tracks with at least one tag. The number of tags decreased significantly from 595 to 195, while the number of tracks didn't change much — decrease from 55 701 to 55 609. Separated by category, we ended up with 95 genre tags, 41 instrument tags, and 59 mood/theme tags.

We generate the standardized splits for training, validation, and testing to foster reproducibility. To avoid the artist and album effects (Flexer and Schnitzer, 2009), we make sure that the tracks from the same artist do not appear in the other subset. To ensure the balanced representation of all tags, we constrain each tag to be represented by at least 40 tracks and 10 artists in the training subset and 20 tracks and 5 artists in validation and testing subsets. We generated 5 splits in this manner to reduce the possibility of bias from a particular split. Some tags were challenging to split properly during generating the splits, so they were discarded. The splitting resulted in 87 genre tags, 40 instrument tags, and 56 mood/theme tags — 183 tags in total, 12 tags being discarded. The eventual number of tracks is 55 525, down from 55 609.

For each split, we also provide the lists of tracks sets per category of tags: genre, instrument, or mood/theme, and for the top 50 tags by the number of tracks (31 genre, 14 instrument, 5 mood/theme tags). The number of tracks in categories is listed in Table 2.2.

The dataset, detailed statistics, pre-processing scripts, and baseline

Figure 2.1: Histogram of top 20 tags of each category

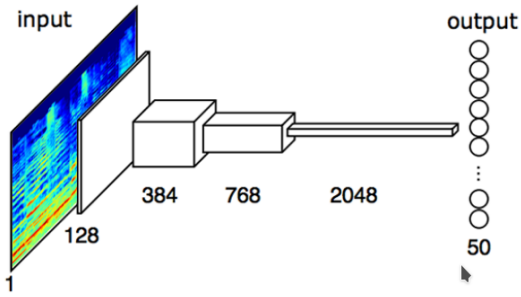



Figure 2.2: FCN-5 architecture (taken from Choi et al. (2017))

implementation are available online.[8] The metadata is released under the CC BY-NC-SA 4.0 license, while the audio files are available under their original Creative Commons licenses.

## 2.4 Baseline models

While we will talk more in detail about the deep learning architectures for auto-tagging in Chapter 3, in this section, we will briefly mention the baseline architecture that was trained for the release of the MTG-Jamendo dataset and its performance.

The baseline architecture is based on the *fully-convolutional network* (FCN) architecture by Choi et al. (2016), more precisely on FCN-5 that has 5 layers that use $3 \times 3$ convolutional units, uses max-pooling and batch normalization after each layer, and has one dense layer with a dropout of 0.5 as the last layer. We have constrained the sizes of the feature layers to make the network smaller, as indicated in Table 2.3.

The original hyper-parameter values were used without modification. The input size is 29.1 seconds of the audio. With the centered sampling strategy, the audio is taken from the middle of the track and random sampling — randomly from any part of the track. All models were trained for

[8] `mtg.github.io/mtg-jamendo-dataset`

| Layer | Original | Adjusted |
|---|---|---|
| 1 | 128 | 64 |
| 2 | 256 | 128 |
| 3 | 512 | 128 |
| 4 | 1024 | 128 |
| 5 | 2048 | 64 |

Table 2.3: Layer sizes of our baseline compared to FCN-5

| Subset | ROC-AUC | PR-AUC |
|---|---|---|
| Genre | **.8337** $\pm$ .0039 | .1522 $\pm$ .0050 |
| Instrument | .7284 $\pm$ .0149 | .1643 $\pm$ .0122 |
| Mood/theme | .7207 $\pm$ .0101 | .1111 $\pm$ .0156 |
| All | .7856 $\pm$ .0077 | .1108 $\pm$ .0069 |
| Top-50 | .7970 $\pm$ .0059 | **.2463** $\pm$ .0073 |

Table 2.4: Baseline performance (random sampling)

100 epochs with ADAM (Kingma and Ba, 2015) optimizer using binary cross-entropy loss and batch size of 64. We use *area under curve of receiver operating characteristic* (ROC-AUC) (Fawcett, 2006) as a primary metric during training. Thus, we used the model with the best validation ROC-AUC to compute the results on the test set. We use PyTorch Lightning[9] framework to implement the training code, which is available in the GitHub repository.[10] To facilitate the reproducibility, we used the seed of 0 to report the results. One split took 12.5 hours to train and test (all categories) on Nvidia 2080Ti with spectrograms stored on SSD as 16-bit floats. In total, all 5 splits took 62.5 hours to process.

The performance of the baseline model averaged over 5 splits (mean $\pm$ standard deviation) is shown in Table 2.4. As the tag distribution is

[9] `pytorchlightning.ai`

[10] `github.com/philtgun/mtg-jamendo-baseline`

quite imbalanced in the dataset (see Figure 2.1), ROC-AUC can give an over-optimistic scores (Davis and Goadrich, 2006), so we also report *area under precision-recall curve* (PR-AUC) as a better metric in our case. The values reported are macro-averaged — averages of tag-wise ROC-AUC and PR-AUC. PR-AUC is quite low, as sparse tags report inferior performance. As the architecture was optimized for the top 50 tags, we can see that it exhibits the highest PR-AUC.

## 2.5 Usage, impact and limitations

After MTG-Jamendo has been introduced in 2019, it has been used in MediaEval[11] challenge *Emotion and theme recognition in music using Jamendo* (2019–2021), which is covered in Chapter 3. It has also been used in multiple works that deal with music auto-tagging:

- Won et al. (2020b) perform analysis of state-of-the-art deep learning auto-tagging architectures on three datasets: MSD, MTAT, and MTG-Jamendo
- Zhao and Guo (2021) introduces the new transformer-based architecture for auto-tagging and uses MTG-Jamendo in addition to other datasets for training and evaluation.
- Ferraro et al. (2021a) introduces a contrastive learning model to learn the music representations and use MTG-Jamendo auto-tagging as one of the evaluation tasks.

The most significant advantage of the MTG-Jamendo dataset is the availability of royalty-free audio. Even for large datasets that provide commercial music such as MSD, given that available audio is usually 30-second previews, having full tracks contributes to a considerable amount of audio data that can be used for self-supervised or semi-self-supervised learning (3 777 hours of MTG-Jamendo vs. 4 210 hours of MSD).

[11] `multimediaeval.github.io`

The commonly mentioned limitation of the MTG-Jamendo dataset is the nature of music that is distributed under a Creative Commons license. It is stylistically different from the typical commercial music. While the production quality is often comparable with commercial music, the fact that independent artists mostly create it imposes a bias on the content of the music. Moreover, because Jamendo is based in Luxembourg, the music is biased toward western traditions, resulting in less representation of the music from other parts of the world and other cultures.

# Chapter 3

# AUTO-TAGGING ALGORITHMS

## 3.1 Introduction

In Chapter 2 we introduced commonly used datasets for music auto-tagging and presented a new dataset: MTG-Jamendo (Bogdanov et al., 2019). While we briefly mentioned the baseline architecture used to provide the baseline performance, we will talk about the state-of-the-art deep architectures for auto-tagging and music emotion recognition in this chapter.

## 3.2 Background

In Section 1.1 we talked about the recommendation systems and the two main approaches: collaborative filtering (CF) and content-based (CB). As we mentioned, CF suffers from several problems, like popularity bias and cold-start problem. In Section 1.3 we have identified that music tags such as genres, moods, and themes are the primary ways for music exploration. As manual annotation of music is quite costly, auto-tagging and classification systems are desirable in music information retrieval.

In a survey of classification approaches by Fu et al. (2011) the classical machine learning approaches were used a lot: feature engineering together with classifier. One of the first examples is the GTZAN paper (Tzanetakis, 2001) where the authors, apart from introducing the dataset, used it for automatic genre classification with crafted features that were supposed to capture timbre, pitch, and rhythm. On top of these features, a k-means classifier and Gaussian mixture models were trained to do the classification.

Once auto-tagging became a topic of broad interest, many more sophisticated methods at that time were based on tag propagation in similarity spaces (Sordo, 2012). Since its inception in 2005, the *Music Information Retrieval Evaluation eXchange*[1] (MIREX) had a task "Genre classification", in 2007 the "Mood classification" was introduced and since 2008 the "Audio tag classification" became the task for generic auto-tagging. Feature engineering and selection were an important part of the methodologies until deep learning became prominent.

Now in the era of deep learning, the approach shifted toward end-to-end learning (Nam et al., 2019). Most models' inputs take either waveforms or mel-spectrograms as input and are trained end-to-end to learn frontend (feature extraction) and backend (classifier). Won et al. (2020b) did quite a thorough review of the recent research in the area of auto-tagging architectures and evaluated them within the same framework on three datasets: MSD, MTAT, and MTG-Jamendo (see Chapter 2). This section will go through the evolution of state-of-the-art auto-tagging architectures through the recent decade.

Convolutional neural networks (CNN) emerged from image processing as efficient networks for image classification and tagging. The mel-spectrogram audio representation is similar to how our ears process audio signals and can be considered an image. Thus the natural thing was to use CNNs on the mel-spectrograms for audio tagging and classification.

VGG (Simonyan and Zisserman, 2015) is a computer vision architecture that utilized multiple $3 \times 3$ CNN layers for image classification. Inspired by it, Choi et al. (2016) introduced *fully-convolutional network*

[1] `music-ir.org/mirex`

| Architecture | Input size | ROC-AUC[1] | PR-AUC[1] |
|---|---|---|---|
| FCN | 29.1s | 0.8255 | 0.2801 |
| CRNN | 29.1s | 0.7978 | 0.2358 |
| MusiCNN | 3s | 0.8226 | 0.2713 |
| Short-chunk CNN | 3.69s | 0.8324 | 0.2976 |
| SampleCNN | 3.69s | 0.8208 | 0.2742 |
| Harmonic CNN | 5s | 0.8322 | 0.2956 |

Table 3.1: Auto-tagging architectures
[1] as reported on MTG-Jamendo by Won et al. (2020b)

(FCN) that consisted of the frontend of 4 layers of CNN and a backend of one dense layer. Many network variations are mentioned with a different number of CNN layers, but the core concept remains the same. The network was designed for the experiments on the MTAT dataset, matching the input size to the audio length and predicting the top 50 tags. Even with the recent advances in the research, this architecture remains competitive and, in its simplicity, provides a substantial baseline — that is why we use a variation of it as a baseline for the MTG-Jamendo dataset (Section 2.4).

In the latter paper, Choi et al. (2017) introduced the *convolutional RNN* (CRNN) that is based on their previous FCN architecture, but with an RNN layer at the end to take advantage of temporal information and reduce the number of parameters of the model. They conclude that CRNN performs comparatively to FCN with a much smaller number of parameters.

Directly adapting multiple computer-vision architectures including VGG, Hershey et al. (2017) evaluated classification performance of multiple architectures on AudioSet dataset (Gemmeke et al., 2017). We refer to the architecture used by Hershey et al. as *VGGish*. It is widely used as an embeddings extractor for audio, as its output represents AudioSet taxonomy and is useless for music auto-tagging.

As FCN and CRNN have quite a large input size that was designed for an MTAT dataset, the *VGG* is the name for architectures that follow a

very similar approach but take much shorter audio input. Those are also called short-chunk CNNs to distinguish them from FCN. In the context of auto-tagging, the reference implementations are considered to be *VGG* by Pons and Serra (2019) and *Short-chunk CNN* by Won et al. (2020b). The final output decisions for the music tracks are usually aggregated by averaging or majority vote over the predictions from individual chunks.

*MusiCNN* (Pons et al., 2018) is a musically-motivated CNN. Instead of square filters, it utilizes vertical and horizontal filters to capture timbre and temporal information. The input size is the same as VGG — 3 seconds. The motivation for the design of this architecture was to take advantage of the domain knowledge, and indeed, it does converge faster. Another approach that attempts to take advantage of domain knowledge is *HarmonicCNN* (Won et al., 2020a), which utilizes a trainable frontend that can exploit the harmonic structure of audio and music.

Apart from the architectures that use mel-spectrograms as input representations, *SampleCNN* (Lee et al., 2017) is an architecture that takes audio waveform as input. It utilizes long 1D CNN blocks at the waveform level with multiple layers on top to aggregate information.

Much more recently, with the popularity of transformer architecture (Vaswani et al., 2017) in natural language processing, Won et al. (2021) has proposed an auto-tagging architecture that uses CNN together with transformer. The authors use the semi-supervised learning approach to train the model, but even fully supervised, the proposed model outperforms previous architectures in music auto-tagging.

In the later chapters of this thesis (Chapters 4, 5), we will refer to FCN, MusiCNN, VGG, and VGG-like (short-chunk) networks as they are quite widely cited and used as a solid basis for more modern architectures. Moreover, they do not require much computational power to train because of their low complexity.

While generic auto-tagging and music classification are popular tasks in music information retrieval, the question remains if some architectures are better than others in recognizing a particular category of tags. As we mentioned in Chapter 2, in the MTG-Jamendo dataset, we have three categories for tags: genres, moods/themes, and instruments. In particular,

mood/theme tags are the second most used tags for music exploration and discovery, according to our survey (see Section 1.5). Thus, with a particular interest in moods/themes, we organized a challenge to focus on improving the state-of-the-art mood/theme auto-tagging.

## 3.3 Music emotion recognition

Music emotion recognition (MER) field is a young and prominent area of research in MIR. Firstly, we need to mention that there is a difference between *perceived* and *induced* emotions in music (Gomez-Canon et al., 2021; Yang and Chen, 2011). Because induced emotions are much more difficult to predict, as they need to account for the user's state and context, MER typically deals with perceived emotions. Secondly, there are two typical approaches to categorize emotions: use categorical labels (happy, sad) or continuous *arousal-valence* space (Russell, 1980). There are benefits and limitations to each of those. In this thesis, we also limit ourselves to perceived emotions and self-reported categorical annotations by creators or curators.

For the categorical characterization of emotions specifically for music, Zentner et al. (2008) developed the Geneva Emotional Music Scales (GEMS). GEMS-45 contains 45 labels that proved to be consistently chosen for describing musically evoked emotive states across a relatively wide range of music and listener samples. There are also condensed versions of the full GEMS scale: GEMS-25 and GEMS-9. The labels are grouped into nine emotional scales, which in turn condense into three "superfactors": *sublimity*, *vitality*, and *unease*.

Our task is not the first one to do that in MediaEval, there was a task titled *Emotions in music* in MediaEval 2013–2015 organized by Soleymani et al. (2013); Aljanaki et al. (2014, 2015). The goal of that task was to predict arousal and valence of emotions for music tracks on the fine-grained resolution (1 sec in 2013, 0.5 sec in 2014–2015). Throughout the years, there were also other sub-tasks: predicting arousal/valence for the whole tracks (2013) and designing new audio features for better

performance (2014).

In another previously mentioned evaluation initiative — MIREX MER was also present in the form of *audio mood classification task* (AMC) (Hu et al., 2008a). Organizers have created a dataset that contains 600 tracks split into five mood clusters with 120 tracks per cluster:

1. Rowdy, rousing, confident, boisterous, passionate
2. Amiable, good-natured, sweet, fun, rollicking, cheerful
3. Literate, wistful, bittersweet, autumnal, brooding, poignant
4. Witty, humorous, whimsical, wry, campy, quirky, silly
5. Volatile, fiery, visceral, aggressive, tense, anxious, intense

Thus the goal of the AMC challenge is to classify the audio into one of these 5 clusters. The task was created in 2007 and has been active with various number of participants. Since 2014, a K-pop extension of this task has been introduced and applied to 1894 tracks.

While emotions and moods are concepts that MER specifically focuses on, in MTG-Jamendo dataset we use the name *moods/themes* for a category of tags. Some tags can easily be identified as emotions and connected to existing taxonomies (GEMS, arousal-valence): *happy*, *relaxing*, *sad*, *calm*; while others completely unrelated to emotions: *film*, *slow*, *melodic*, *sport*. However, there are multiple tags that are describing specific moods that are only partially related to emotions: *dark*, *love*, *romantic*, *epic*. Figure 3.1 shows the full list of the mood/theme tags from MTG-Jamendo dataset. We decided not to try to separate tags into emotion and non-emotion-related subcategories, as it is not as straightforward and subjective, thus keeping all of them under the umbrella term *moods/themes*.

## 3.4 MediaEval

To promote the MTG-Jamendo dataset and facilitate the improvements of state-of-the-art auto-tagging of moods/themes, we have been organizing a

challenge titled *Emotion and theme recognition in music using Jamendo* as part of *Multimedia Evaluation Benchmark*[2] (*MediaEval*) initiative in 2019–2021. MediaEval is a community-driven benchmark that is run by the MediaEval organizing committee consisting of the task organizers of all the individual tasks in a given year. MediaEval tasks are largely autonomous, and each team of task organizers is responsible for running their tasks.

### 3.4.1 Task description

Our challenge invites the participants to build an auto-tagging system to predict mood/theme tags of the MTG-Jamendo dataset. We choose the mood/theme subset (instead of instruments or genres) for several reasons. Firstly, we already mentioned the connection to the music discovery and exploration in Section 1.5. Secondly, the mood/theme tags are pretty challenging, as it was shown in Section 2.4. Among all the subsets that we have introduced, this subset has the poorest performance in the baseline (see Table 2.4). Furthermore, thirdly, because of novelty and appeal to the participants, moods/themes are less explored than genre recognition which is a more popular task.

As mentioned in Section 2.3, we provide audio as well as extracted with Essentia mel-spectrograms and features. The histogram of all the tags is shown in Figure 3.1. Participants could use any data as input for their systems. We use PR-AUC as the primary metric and ROC-AUC, precision, recall, and F-score as additional metrics for evaluation. Every team could submit up to 5 runs. Participants were provided with scripts to download data and reference code implementation of the baseline VGGish architecture.

To evaluate the submissions, we asked participants to generate activation values (*predictions*) for the test set (4231 tracks and 56 tags) that were used to calculate PR-AUC and ROC-AUC. We also asked participants to generate binary *decisions* on the test set to measure precision, recall, and F-score. As the primary evaluation metric is PR-AUC, deciding on the

[2] `multimediaeval.github.io`

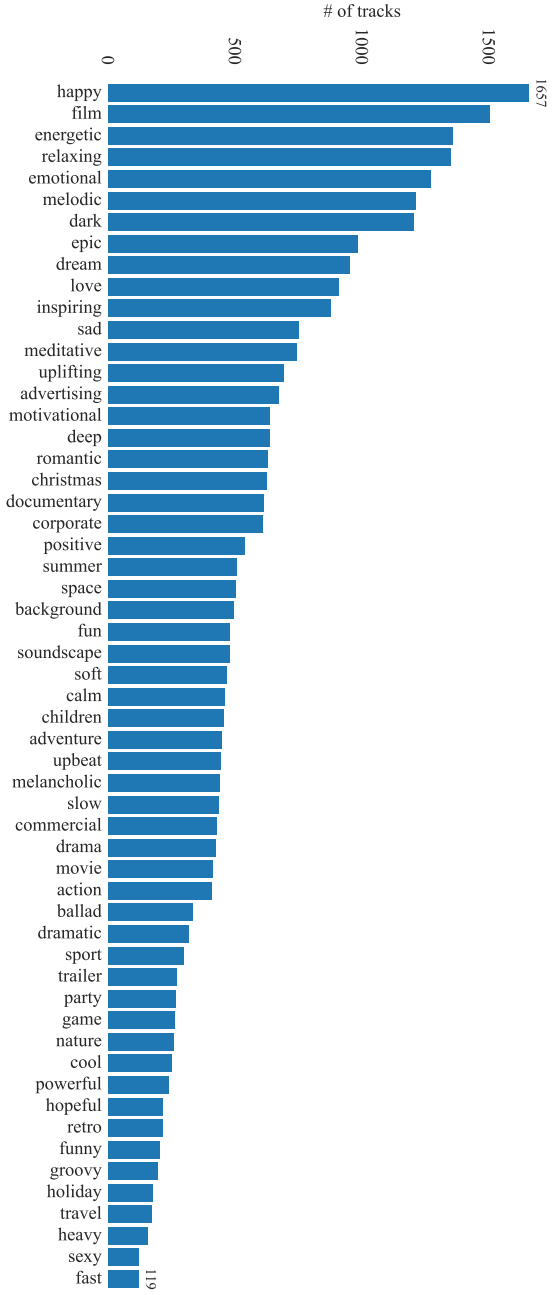



Figure 3.1: Histogram of all mood/theme tags

| Year | 2019 | 2020 | 2021 |
|---|---|---|---|
| Registered | 14 | 12 | 11 |
| Submitted results | 8 | 6 | 4 |
| Submitted working notes | 6 | 6 | 4 |

Table 3.2: Number of teams participating

thresholds to transform predictions into decisions is not the focus of the task. Thus, we provided the script to do the thresholding and generate binary decisions to maximize F-score.

The statistics of the number of participating teams and submissions throughout the years are presented in Table 3.2. In the following sections, we will go through each year of the task, the submissions, and their contributions.

### 3.4.2 2019 edition

For the 2019 edition of the task, out of 8 teams that submitted their runs, 6 teams have submitted the working notes describing their approach (see Table 3.2). The leaderboard summarizing the approaches and performances is presented in Table 3.3. More detailed results are available online[3].

The team that achieves the highest performance — CP-JKU (Koutini et al., 2019) applies multiple techniques throughout the whole process. They use ResNet (He et al., 2016) as a basis for the architecture together with techniques from acoustic scene classification such as receptive field (RF) regularization, frequency-aware (FA) CNN layers, and shake-shake regularization. For the data augmentation, the Mixup technique is used. Multiple averaging techniques are used: stochastic weight averaging during training, snapshot averaging for testing, and multi-model averaging for ensemble results. This team focused on optimizing the architecture and training process to get the highest score, which they achieved.

[3] `tinyurl.com/mediaeval2019music`

| | Team | Run | AUC | |
|---|---|---|---|---|
| | | | **PR** | ROC |
| 1 | CP-JKU | Ensemble | .1546 | .7729 |
| 2 | CP-JKU | ShakeFaResNet | .1480 | .7716 |
| 3 | CP-JKU | FaResNet | .1463 | .7574 |
| 4 | AMLAG | MobileNetV2+Att | .1258 | .7528 |
| 5 | YL-UTokyo | CNN 6L | .1255 | .7531 |
| 6 | AMLAG | MobileNetV2 | .1183 | .7324 |
| 7 | AugLi | Ensemble † | .1174 | .7424 |
| 8 | CP-JKU | CRNN | .1171 | .7380 |
| 9 | AIT-DIL* | 01 | .1126 | .7191 |
| 10 | TaiInn (In) | RndS | .1103 | .7186 |
| 11 | TaiInn (In) | RndS+Att | .1103 | .7230 |
| 12 | **baseline** | VGGish | .1077 | .7258 |
| 13 | Taiinn (Tw) | Fx-VQVAE1+CNN ‡ | .1076 | .7207 |
| 14 | AugLi | All DS † | .1038 | .7260 |
| 15 | Taiinn (Tw) | FX-VQVAE1+GRU ‡ | .1037 | .7140 |
| 16 | CP-JKU | ResNet34 | .1020 | .7168 |
| 17 | AugLi | All CRNN † | .0999 | .7066 |
| 18 | Taiinn (Tw) | VQVAE1+CNN ‡ | .0994 | .7146 |
| 19 | Taiinn (Tw) | VQVAE1+GRU ‡ | .0984 | .7103 |
| 20 | AugLi | All DS 5s † | .0980 | .7162 |
| 21 | AugLi | All DS 1s † | .0972 | .7146 |
| 22 | TaiInn (In) | RndS+Ess | .0897 | .6852 |
| 23 | TaiInn (In) | RndS+Att+Ess | .0891 | .6839 |
| 24 | Taiinn (Tw) | VQVAE2+GRU ‡ | .0860 | .6916 |
| 25 | TaiInn (In) | BsS | .0795 | .6998 |
| 26 | MCLAB-CCU* | 03 | .0341 | .5014 |
| 27 | MCLAB-CCU* | 01 | .0332 | .4891 |
| 28 | MCLAB-CCU* | 02 | .0332 | .4937 |
| 29 | **baseline** | popular | .0319 | .5000 |

Table 3.3: MediaEval 2019 leaderboard

* Teams that didn't submit working notes paper

Used external data: † Audioset, ImageNet; ‡ MSD, MTAT


Figure 3.2: Not all tags benefit from joint training (taken from Sukhavasi and Adapa (2019))

The team with the next best result is AMLAG (Sukhavasi and Adapa, 2019). Their architecture of choice is MobileNetV2 (Sandler et al., 2018) with addition of self-attention. Data augmentation techniques such as Mixup and SpecAugment were also used in this submission. The main insight from this submission is that this team took time and effort to investigate the individual tag losses during training and discovered that some tags' performance suffers from joint training (see Figure 3.2). However, the attempts to find several groups of tags that could benefit from training together ultimately lead to poorer overall results. They only employ the early stopping for the individual tags based on the individual loss. Moreover, the authors report the approaches that did not work, which is valuable.

YL-UTokyo (Yi et al., 2019) utilize the simple approach and augment the baseline model provided from 5 convolutional layers to 6 convolutional layers using ELU. They achieve the performance that puts them as the third-best team.

AugLi (Amiriparian et al., 2019) utilize CRNN that consists of an AudioSet-pretrained VGGish model with the final pooling layer replaced by RNN. The second parallel model in the framework is DeepSpectrum — ImageNet-pretrained VGGish model, taking 1- and 5-second chunks of audio as input and feeding the penultimate feature vectors into RNN. Authors test LSTM, GRU, and BLSTM as RNN units and ensemble them together in different configurations as submissions.

Two teams TaiInn from Innsbruck and Taiwan collaborated to work on the challenge and submitted different approaches. TaiInn (Innsbruck) (Mayerl et al., 2019) uses CNN feeding into GRU units with dense layers in the end with ELU activation. Authors also test feeding Essentia features as input to the last dense layers in some variations and adding attention mechanism after GRU layers. The model was trained only for 16 epochs, thus achieving competitive performance with little computational resources. TaiInn (Taiwan) (Hung et al., 2019) utilize different approach based on using two variants of VQ-VAE as a feature extractor (pre-trained of MSD) and comparing two different classifiers: GRU and CNN (using MTAT as additional data).

To summarize this first edition of the challenge, it is essential to note that the winning approach did not use any external data or pretrained model. However, the team spent much effort on data augmentation, regularization, and optimization. The AMLAG submission is much more insightful, as they show that some tags are difficult to train.

### 3.4.3 2020 edition

In 2020 6 teams submitted runs and working notes (see Table 3.2). We provide the summary of the performances in the Table 3.4 and approaches below (details are available online[4]).

SAIL-MiM-USC (Knox et al., 2020) use modified short-chunk CNN architecture (Won et al., 2020b) as well external data from Music4All dataset that exactly matches provided tags to expand the training set. Moreover, they use MSD to pre-train the lower layers of the network.

[4] `tinyurl.com/mediaeval2020music`

| | Team | Run | AUC **PR** | AUC ROC |
|---|---|---|---|---|
| 1 | SAIL-MiM-USC | Ens AllData † | .1609 | .7812 |
| 2 | SAIL-MiM-USC | Focal+AllData † | .1561 | .7782 |
| 3 | **Best 2019** | Ensemble | .1546 | .7729 |
| 4 | SAIL-MiM-USC | Ens Jamendo | .1421 | .7625 |
| 5 | HCMUS | WN-EffB7+M-EffB0 ‡ | .1414 | .7663 |
| 6 | HCMUS | WN-MobV2+M-EffB0 ‡ | .1413 | .7680 |
| 7 | HCMUS | Mel-EffB0 ‡ | .1398 | .7627 |
| 8 | AugsBurger | Fusion+AUGment | .1313 | .7533 |
| 9 | UAI-CNRL | ResNet34+Att | .1275 | .7360 |
| 10 | AugsBurger | CBAMs-fusion | .1227 | .7405 |
| 11 | AugsBurger | CBAM-GRU-256 | .1203 | .7394 |
| 12 | AUGment | AReLU+Att+VGGish | .1178 | .7353 |
| 13 | AUGment | AReLU+Att | .1136 | .7323 |
| 14 | AUGment | Att | .1082 | .7169 |
| 15 | **baseline** | VGGish | .1077 | .7258 |
| 16 | AUGment | AReLU | .1072 | .7281 |
| 17 | AugsBurger | CBAM-GRU-128x2 | .1070 | .7158 |
| 18 | HCMUS | WN-EffB7 ‡ | .1054 | .7185 |
| 19 | UIBK-DBIS | A-CRNN | .0965 | .7043 |
| 20 | UIBK-DBIS | A-ECRNN-F1 | .0903 | .6849 |
| 21 | UIBK-DBIS | A-ECRNN-Man2 | .0900 | .6885 |
| 22 | UIBK-DBIS | ECRNN-Man2 | .0887 | .6953 |
| 23 | UIBK-DBIS | A-ECRNN-Man3 | .0862 | .6829 |
| 24 | **baseline** | popular | .0319 | .5000 |

Table 3.4: MediaEval 2020 leaderboard
Used external data: † MSD, Music4Aall; ‡ NSynth

However, the most significant contribution is the training with different loss functions: focal loss, class-balanced loss, distribution-based loss, and ensembling of the resulting models. While models that take advantage of external data exhibit the highest performance, the model trained without external data still performs better than other submissions, although not enough to beat the best submission of 2019.

HCMUS (Do et al., 2020) utilize WaveNet-style autoencoder pre-trained on NSynth dataset as a feature extractor in conjunction with mel-spectrograms. WaveNet features are fed into MobileNetV2 and EfficientNet-B7, and mel-spectrograms — to EfficientNet-B0. SpecAugment was used for data augmentation. The authors also attempted to reduce multiple labels to a single label. Different combinations of those have been submitted as runs with the ensemble of WaveNet features and mel-spectrograms achieving the highest score. The team also reported trying SVM, InceptionNet, ResNet, and self-attention, which did not improve the performance.

AugsBurger (Gerczuk et al., 2020) use CRNN framework with mel-spectrograms feeding into ResNet with Convolutional Block Attention Modules (CBAMs) and GRU/LSTM blocks in the end. Authors perform many fusion experiments of different architecture versions and fuse with another approach from team AUGment.

AUGment (Rajamani et al., 2020) introduce self-attention into the provided VGGish baseline. They replace various layers of the CNN with stand-alone self-attention, thus reducing the number of parameters that do not deteriorate performance much. Authors also experiment using AReLU in all layers of CNN.

UAI-CNRL (Dipani et al., 2020) utilize ResNet34 as feature extractor that feeds into self-attention module. Multiple data augmentation techniques are used: Mixup, SpecAugment, random cropping, and scaling. The team only submitted one run, so no architecture variations were explored.

UIBK-DBIS (Vötter et al., 2020) encompasses some participants from TaiInn (Innsbruck) team from 2019. Their 2020 submission uses a slight adaptation of their previous CRNN model and a CNN-based model with

ELU activations. The CNN-based model did not perform well, so the submission only included CRNN-based approaches. However, the important contribution of this team is that they attempt to separate tags into groups to train separate models and ensemble them for the final predictions. Three strategies are: linear (based on lexicographic order), performance (based on F1 and PR-AUC), and moods vs. themes (vs. uncertain, judged by four human judges). The dataset was also augmented to provide more training samples for underrepresented tags based on the tag frequency.

### 3.4.4 2021 edition

In 2021 only 4 teams managed to submit runs and working notes (see the results in Table 3.5). We do not include the best runs from previous years in this year's leaderboard, as they have not been beaten. More details are available online.[5]

Team lileonardo (Bour, 2021) achieved the highest performance this year. The main contribution is to try and employ frequency-dependent convolutional layers. Several aspects were investigated: mel-spectrogram resolution (96 and 128); input length (128 and 224 frames); and loss functions (BCE, weighted BCE, and focal). The submitted runs consisted of ensemble models for traditional convolutions, frequency-dependent ones, and everything together. Although frequency-dependent convolutions do not perform much better than traditional ones, the performance improves from ensembling many models. The best single model uses 96 mel bands and weighted BCE loss and input length of 224 frames and achieves a PR-AUC of 0.1447.

SELAB-HCMUS (Pham et al., 2021), participating second year in the row used the co-teaching (Han et al., 2018) paradigm to train EfficientNetB0 and ReXNet in parallel. Similar to the previous year's submission (Do et al., 2020), they used single labels per track instead of multi-labels similarly, as well as Mixup and SpecAugment.

Team Mirable (Tan, 2021) tried to use a semi-supervised approach of noisy student methodology (Xie et al., 2020) to take advantage of the

[5] `tinyurl.com/mediaeval2021music`

| | Team | Run | AUC | |
|---|---|---|---|---|
| | | | **PR** | ROC |
| 1 | lileonardo | Ens all | .1508 | .7747 |
| 2 | lileonardo | Ens Freq-Dep | .1478 | .7703 |
| 3 | lileonardo | Ens Convs | .1468 | .7690 |
| 4 | SELAB-HCMUS | Ensemble | .1435 | .7599 |
| 5 | SELAB-HCMUS | EffNet Co-Teach | .1415 | .7574 |
| 6 | Mirable | Ensemble † | .1356 | .7687 |
| 7 | SELAB-HCMUS | ReXNet Co-Teach | .1343 | .7504 |
| 8 | Mirable | Short HPCP | .1275 | .7541 |
| 9 | SELAB-HCMUS | ReXNet | .1261 | .7463 |
| 10 | Mirable | Long HPCP Noisy † | .1235 | .7613 |
| 11 | UIBK-DBIS | Ens VGGish k-means | .1087 | .7046 |
| 12 | **baseline** | VGGish | .1077 | .7258 |
| 13 | UIBK-DBIS | Ens VGGish dk-means | .0984 | .6829 |
| 14 | UIBK-DBIS | Ens ResNet linear | .0921 | .6996 |
| 15 | UIBK-DBIS | Ens ResNet k-means | .0910 | .6916 |
| 16 | UIBK-DBIS | Ens ResNet dk-means | .0799 | .6807 |
| 17 | **baseline** | popular | .0319 | .5000 |

Table 3.5: MediaEval 2021 leaderboard
† used full MTG-Jamendo dataset as external data

rest of the MTG-Jamendo dataset that is not tagged with moods/themes. Their architecture is based on CRNN with added residual connections and GeMPool instead of MaxPool. Besides mel-spectrograms, they also used computed HPCPs as additional input to their model, which improved performance. Authors have also tried a relatively long input length of 185 secs and 9.25 secs. The results of noisy student training are inconclusive, presumably due to the abstract nature and subjectivity of emotion and theme labels.

UIBK-DBIS (Mayerl et al., 2021) participated for the 3rd year in a row. This year, their approach was to automatically cluster tags based on similarity or dissimilarity, train individual models on the clusters, and use the ensemble for prediction. Tag clusters were calculated based on high-level Essentia features, and the authors propose a variation of k-means — dissimilar k-means (dk-means) to calculate clusters that gather the most different tags. The architectures submitted included VGG and ResNet-18.

### 3.4.5 Summary

There were a lot of different architectures being used in the task — mostly CNN-based (VGGish, ResNet, MobileNet, EfficientNet), and WaveNet, VQVAE, as feature extractors. Many teams added RNN units at the end of the CNN pipeline, constructing the CRNN framework. Typical RNN units used were GRU, LSTM, and BLSTM.

Data augmentation was commonly used among participants: Mixup, SpecAugment, transformations (cropping, scaling). As the dataset is imbalanced, tag frequency aware augmentations were also used to increase the number of samples for underrepresented tags.

Usually, binary cross-entropy was used as a loss function. However, other types of losses, such as focal, class-balanced, and distribution-balanced losses, are also present. Ensemble models and late fusion was used almost by all participants as an easy way to improve the performance of the models.

### 3.4.6 Per-tag performances

With 16 teams and 49 submissions over two years of the challenge, we wonder about the performance of the individual tags - to see if any tags, in particular, were easy or challenging or if the performance is directly correlated with the number of tracks available for training. Figure 3.3[6] shows the PR-AUC performances of the individual submissions per individual tag. On the X-axis, 56 tags are shown in the order of decreasing the number of tracks in the training set (left to right). The Y-axis represents all submissions (from all years) ranked by average PR-AUC performance.

Interestingly enough, different architectures have similar performance for the same tag, while performances vary depending on the tag. While the expected result would be the higher performance of the tags on the left (with more tracks in the training set) that would slowly decrease towards the right, some tags clearly stand out and are much easier to predict than others. Some tags that stand out: *deep, summer, children, corporate*. On the left side of the figure, while *happy, energetic and relaxing* have a slightly higher number of tracks, *film, dark and epic* tend to have higher performance.

Interestingly, models that have been pre-trained on other datasets do not perform much better on the difficult tags. However, they gain some performance due to better predictions on the non-difficult tags. Here are some interesting observations of differences in performance between the submissions:

- HCMUS team (2020) have considerably lower performance on the *advertising* tag compared to other high-performing approaches using WaveNet-style encoder pre-trained on NSynth, mel-spectrograms with EfficientNet, as well as the ensemble model.

- YL-UTokyo's (2019) quite simple approach of increasing the depth of convolutional units from 5 to 6 has higher performance in a lot of tags (including difficult ones: *uplifting, background, adventure,*

[6]Interactive version available at
`philtgun.me/mediaeval-emothemes-explorer`

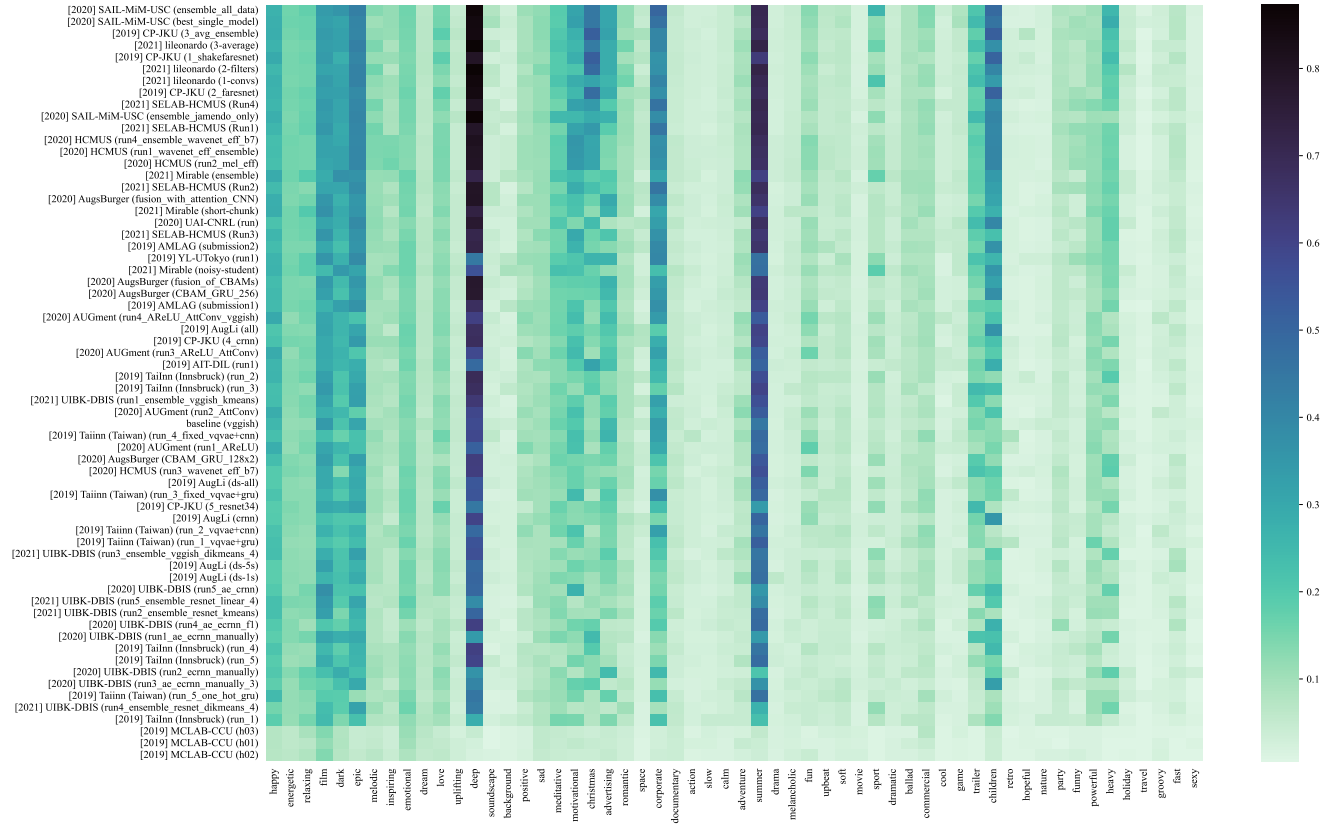



Figure 3.3: Per-tag PR-AUC performances

*soft, hopeful, funny*, etc.) than the best teams. That came at the cost of significantly lower performance on the "easy" tags (deep: .44) — ones that other teams have achieved relatively high performance on, thus lowering the overall average team performance.

- Similar behavior can be observed in team Mirable (2020), who used simple CRNN with residual connections and GeMPool with noisy student training on the full MTG-Jamendo dataset. While *deep* and *summer* have lower performance, a lot of more difficult tags (*relaxing, meditative, inspiring*, etc.) have higher PR-AUC than the top submissions

- While the *retro* tag is quite difficult to predict (PR-AUC of 0.035 among top teams), team TaiInn (Taiwan), with their VQ-VAE approach, managed to get PR-AUC of 0.057–0.087

### 3.4.7 Dataset imbalances

Looking deeper into the reasons behind the trends of difficult and easy tags, it becomes evident that even if the distribution of tags in the dataset is quite unbalanced, it also is slightly different across train, validation, and test sets (see Figure 3.4). The hypothesis that more variance in the training data leads to better performance was not supported well enough. Tags such as *love*, *sad*, *melancholic*, *romantic* and *space* that have higher number of artists using them on average don't exhibit considerably better performance in the submissions.

Another potentially interesting effect to investigate was the impact of tags with a higher number of tracks in the test set than other tags. Some examples include *film*, *emotional*, *children*, *commercial*. Tag *children* indeed exhibits increased performance compared to the tags with a similar number of tracks in the training set, but that is the only occurrence.

We also took a closer look at which tags are usually used together. To measure the co-occurrence, we counted the number of tracks where the pair of tags was used together. Some co-occurrences that are noticeable by the absolute number of tracks (see Figure 3.5):

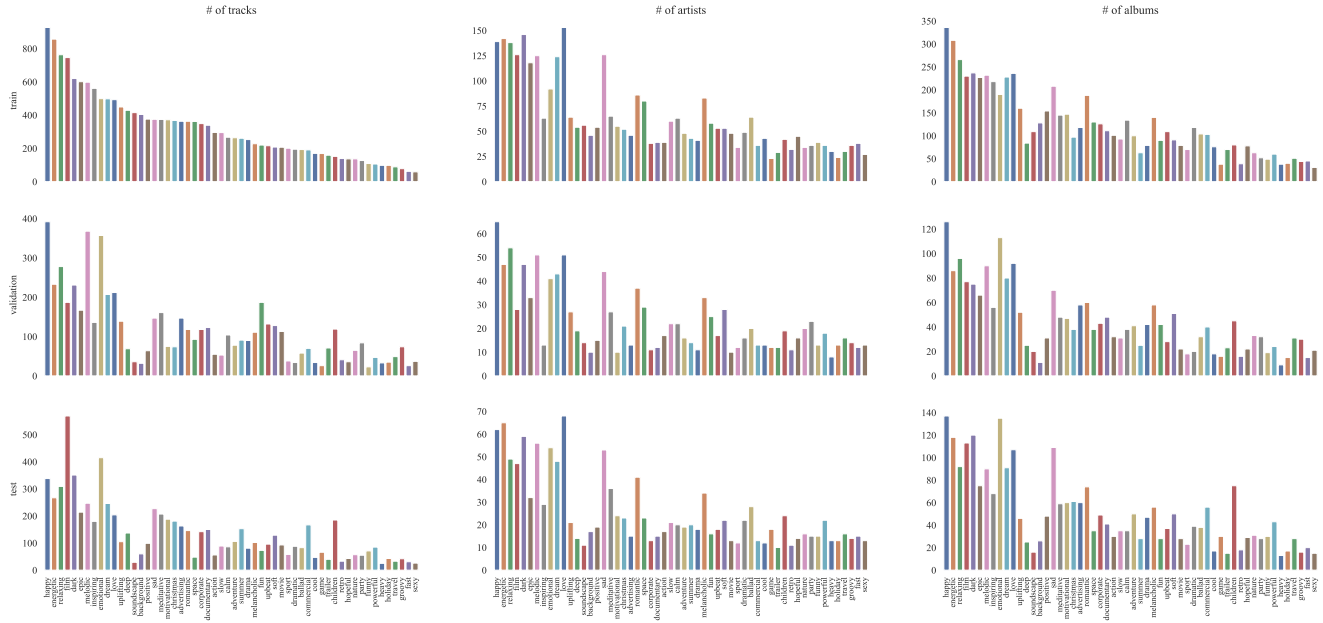



Figure 3.4: Number of tracks, artists and albums per tag in train, validation and test sets of split-0

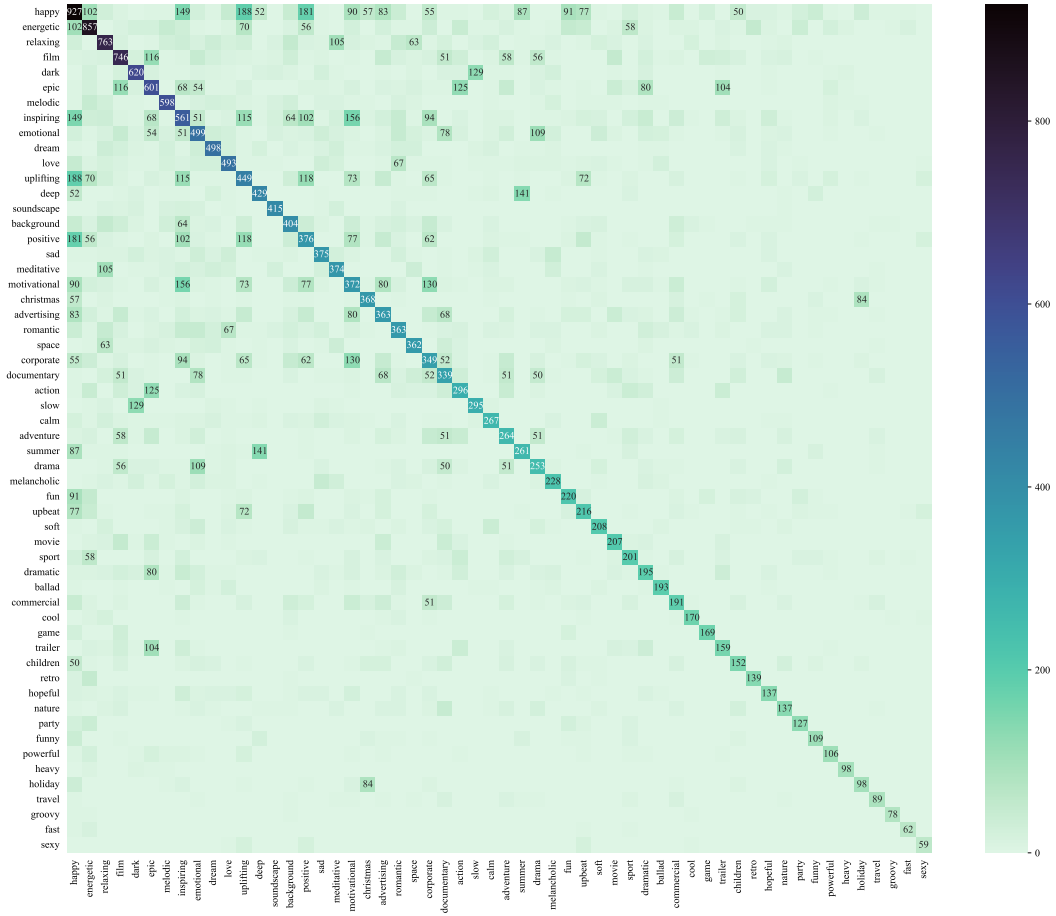



Figure 3.5: Co-occurrence of tags in train set of split-0 in terms of number of tracks

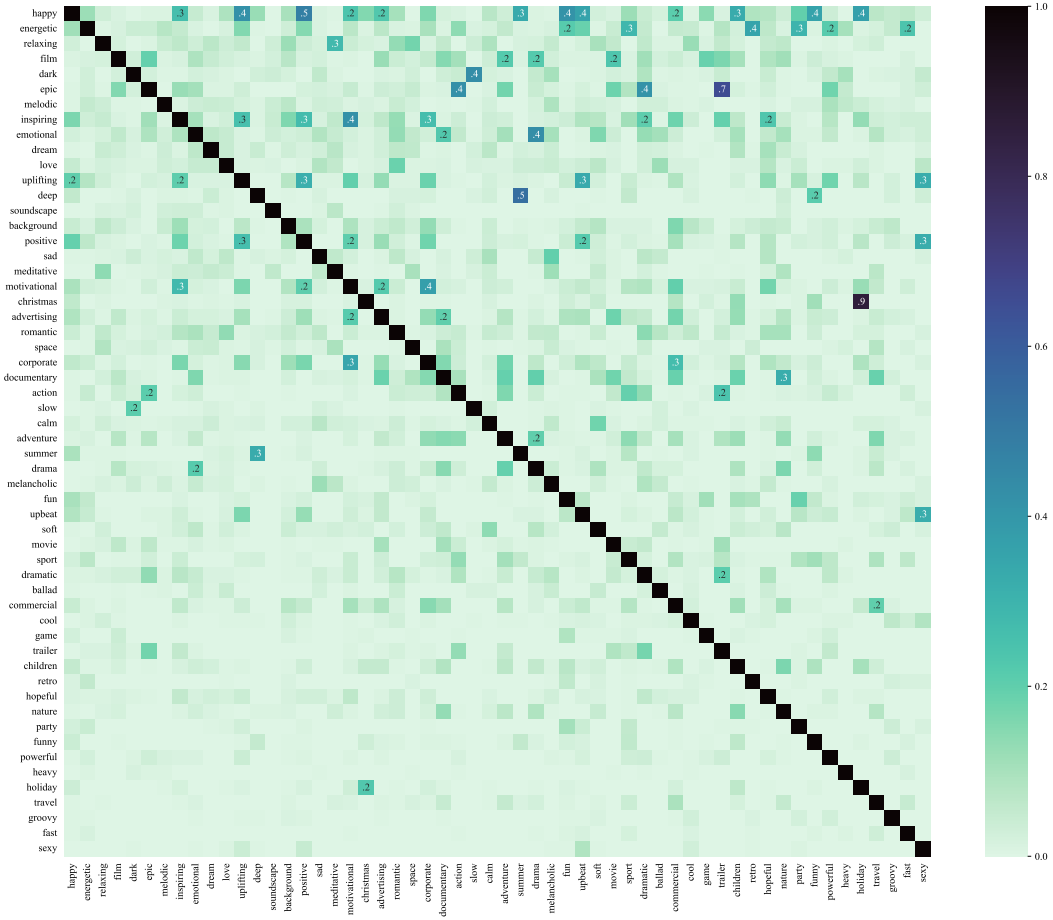



Figure 3.6: Co-occurrence of tags in train sets of split-0 as percentage of total number of tracks (normalized columns)

- happy — uplifting (188)
- happy — positive (181)
- inspiring — motivational (156)
- happy — inspiring (149)
- **deep — summer** (141)
- **motivational — corporate** (130)
- dark — slow (129)
- epic — action (125)
- **epic — trailer** (104)

While the co-occurrences in the tags that already have many tracks are not correlated to high performance, it is more noticeable in the tags that have fewer tracks in the training set (the pairs are highlighted in bold)

However, the high performance of tags *deep* and *summer* cannot be just explained by the co-occurrence, as other co-occurrences do not result in much difference in performance. Upon closer inspection, tag *deep* is mostly used in reference to a genre *deep house*. While some moods and themes might have a predisposition to some genres, usually, there is enough variety within tags. In this case, the subgenre (even more specific) is the same for many annotated tracks, potentially reducing the task to genre identification.

### 3.4.8 Insights from submissions

#### Training on external data

Non-surprisingly the best submission from all three years from SAIL-MiM-USC took advantage of using external data from MSD and Music4All datasets that matched some of the challenges' tags. More data improves the performance, especially in this challenge, where the dataset is not that big.

#### Ensemble of loss functions

In 2020 team SAIL-MiM-USC (Knox et al., 2020), instead of ensembles of different architectures, as it is commonly done among other submis-

sions, hypothesized that the challenge lies in the problematic distribution of the data. They try several loss functions designed to work better with less uniform distributions: focal, class-balanced, and distribution-balanced losses. While they do not beat the best 2019 submission training only on provided data, they come close (PR-AUC of 0.1421 compared to 0.1546).

#### Reducing multi-labels to single labels

In 2020 team HCMUS (Do et al., 2020) have introduced a data-balancing pre-processing step. The idea is to reduce multi-labels to single labels per track, keeping the most important tag, i.e., the one with the least representations. In 2020 the reported increase in performance due to the data balancing improved PR-AUC of 0.127 to 0.134 on their EfficientNet-B0 architecture.

#### Tonal information

Team Mirable (Tan, 2021) hypothesized that including tonal information is helpful for emotion in music tagging, and to support their hypothesis computed harmonic pitch class profiles (HPCPs) and used this data to improve their performance. For long input length (185 sec), adding HPCPs to mel-spectrograms has improved PR-AUC from 0.1024 to 0.1220, and for short (9.25 sec) has not provided much improvement — PR-AUC went from 0.1234 to 0.1275.

### 3.4.9 Conclusion

Thus, the conclusions from three years of organizing this task are that there is possibly a glass ceiling related to the training data from MTG-Jamendo. The distribution is unbalanced, and so far, the high performance was achieved because of the submissions specifically tackling the problem of the unbalanced distribution.

Moreover, the tag co-occurrence also contributes to the high performance of certain tags (deep/summer, motivational/corporate, epic/trailer),

as it leads to some tags having a much higher amount of data to be trained on. The future work is to group some tags into the clusters and equalize the distribution to reduce the impact of non-uniformity.

Many valuable insights and discussions about the architectures are provided in the working notes. However, because of the lack of data, the choice of architecture can lead to better efficiency (faster convergence and lighter models) but not to a significant increase in performance. Nevertheless, it is important to consider efficiency in the deep learning model training; thus, simple models are still valid options. The results align with the study by Won et al. (2020b) with the simple short-chunk VGG architecture with residual connections remaining competitive.

# Chapter 4

# MUSIC SIMILARITY

## 4.1 Introduction

As we discussed in Section 1.1, music recommendation systems are currently one of the primary ways for people to listen to and find music. While collaborative filtering (CF) approaches are still within the state-of-the-art of personalized music recommendations, pure CF falls short of the cold-start problem and non-personalized recommendations. The content-based (CB) approaches can provide recommendations and suggestions based on item-to-item similarity without CF data, and they are commonly used together with CF in modern recommendation systems (Ricci et al., 2022) to solve the cold-start problem. However, when there is no CF data available due to design decisions or privacy concerns, CB approaches are the only ones that can provide recommendations.

In the domain of music, there are different modalities to the content that can be used for CB approaches. Apart from the audio signal, data that can be used is metadata, user-defined tags, reviews, etc. This thesis will focus on audio and current state-of-the-art auto-tagging models. We are interested in how consistent are the latent spaces extracted by auto-tagging models between each other, particularly concerning the choice of the training dataset, architecture, or the layer of the network. These insights can show which variable contributes to the most different results,

which can inform practical decisions on prioritizing models for A/B testing in an industry scenario with limited resources.

Latent similarity spaces are also quite extensively used in music visualization interfaces (Knees et al., 2020) (we will talk more about those in Chapter 5), where such similarity spaces represent music on a 2D plane or 3D space and facilitate exploration, discovery, and re-discovery of music. The latent spaces usually are high-dimensional, and part of the information is lost by performing the projection. We are interested to see how well the commonly used projection methodologies represent and transform similarity space and how much of the nearest neighbors' information is preserved.

Furthermore, we investigate how CB approaches compare to CF approaches in a user-less scenario, with CF factors representing a latent similarity space. The motivation is to see if different CB approaches capture more or less of the information from user interactions in CF systems, thus resulting in more or less similar nearest neighbor results.

## 4.2 State of the Art

Music similarity is a widely researched topic in music information retrieval. In MIREX (Music Information Retrieval Evaluation eXchange), the task of music similarity was active until 2015, as eventually, the performances of the submitted systems reached a glass ceiling stemming from evaluation being subjective and limited inter-rater agreement (Flexer, 2014). Music similarity is relatively subjective as humans use different dimensions for assessing similarity: genre, moods, tempo, instrumentation, etc. Recent work investigates the importance of inter- and intra-rater agreement in the context of music similarity and recommendation (Flexer et al., 2021) that questions the validity of experiments on general music similarity. Thus, it is vital to minimize the ambiguity of the evaluation process and provide context or a scenario to allow users to provide more informed answers instead of asking vague questions about which track is more or less similar to the reference track.

In the context of music recommendation, there are many approaches to solve the cold-start problem (Ricci et al., 2022), for example, deep-learning and hybrid approaches (Oramas et al., 2017a; Wang and Wang, 2014), or ones trying to predict the CF latent factors from audio (van den Oord et al., 2013; Ferraro et al., 2021b). They all attempt to bridge the gap between CF and CB, thus requiring CF data to train the model. This chapter considers the scenario without personalization (anonymous user), i.e., where the system has no information about the user and needs to consider only track-to-track similarity.

Among the visualization interfaces of music collections, there are several commonly used techniques to reduce the dimensionality of the original latent spaces (Knees et al., 2020). One of the first successful techniques is self-organizing maps (SOM) (Kohonen, 2001) used in Islands of Music (Pampalk et al., 2002) and other works that have followed and were inspired by it. We talk more about state of the art in music visualization interfaces in Section 5.2. However, for the context of this chapter, what is essential to know is that in more recent works, the newer algorithms such as t-SNE (Maaten and Hinton, 2008) and UMAP (McInnes et al., 2018) gained popularity. They transform space in a non-linear way attempting to capture the relations between individual elements. The classic non-stochastic principal component analysis (PCA) approach can also be used (Smilkov et al., 2016). While it is not as good at capturing the individual relationships between items, it captures the global structure of the whole space.

## 4.3 Similarity metric

We aim to compare multiple latent spaces that contain the same set of items (music tracks). If we use one track as a reference and retrieve the nearest neighbors to the reference, we would have several different lists of nearest neighbors for each latent space. We introduce a simple metric $S_n$ to calculate the similarity between two ranked lists of nearest neighbors $L$ at the cutoff of $n$ tracks obtained from two music similarity spaces $X$

and $Y$. To differentiate this similarity between spaces from the music similarity that we also talk about, we use the term *NN-similarity* in this thesis. We divide the number of tracks that are common in both lists by the cutoff to obtain the value between 0 (no common tracks) and 1 (all tracks are the same):

$$S_n(X, Y) = \frac{|L_{X,n} \cap L_{Y,n}|}{n} \tag{4.1}$$

If we consider the following example of $n = 5$ nearest neighbors to the track $t_0$ in the spaces $X$ and $Y$, we would calculate the NN-similarity in the following way:

$$\begin{aligned} L_{X,5} &= (t_1, t_2, t_3, t_4, t_5) \\ L_{Y,5} &= (t_2, t_6, t_3, t_7, t_8) \\ L_{X,5} \cap L_{Y,5} &= \{t_2, t_3\} \\ S_5(X, Y) &= 2/5 = 0.4 \end{aligned}$$

$S_n$ does not take into account the ranking: $t_2$ is ranked higher than $t_3$ in both $L_{X,5}$ and $L_{Y,5}$, but even if the relative rank would be reversed for $L_{Y,5}$, $S_5(X, Y)$ would still have the same value. In reality, if the cutoff $n$ is much smaller than the number of tracks in the dataset ($n \in \{5, 10, 100, 200\}$), the primary difference between the lists is the number of intersected elements, not what is the difference between their ranks. The only potential benefit of using metrics that take ranks into account is getting the finer difference between lists with the same amount of common tracks. We tried to use Spearman rank correlation or rank-based overlap (RDO) (Webber et al., 2010), and these metrics did not provide more information about the difference between pairs compared to simple $S_n$.[1]

[1] See additional materials at the companion website `philtgun.me/deep-neighbors` for reports on other metrics.

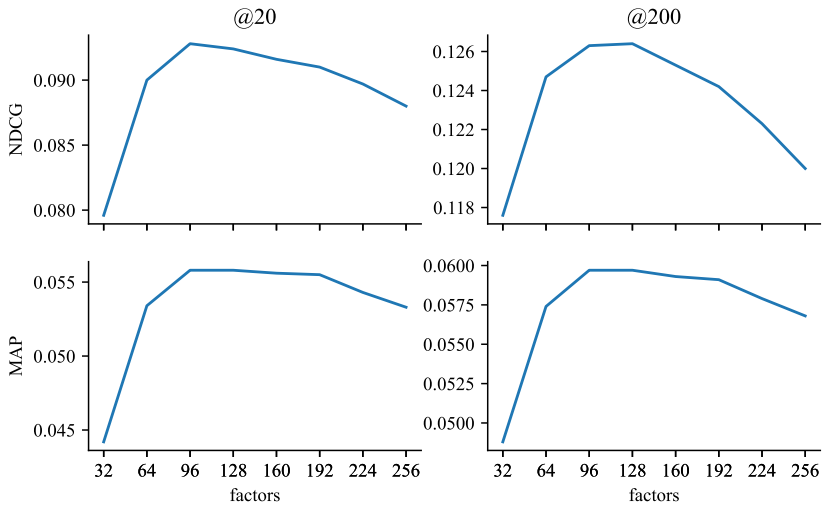



Figure 4.1: Baseline CF evaluation

## 4.4 Data

As we mentioned in Chapter 2, Jamendo[2] is a platform that provides royalty-free music for commercial and personal use, including music streaming for venues or video production. The work introduced in this chapter was partially performed as an internship in Jamendo. We use audio tracks from their complete catalog, not only from MTG-Jamendo (Bogdanov et al., 2019).

### 4.4.1 Collaborative filtering features

The collaborative filtering data was provided as part of the collaboration with Jamendo and included 2.2 million interaction events (including plays, skips, etc.) that have associated numeric values assigned via an internal system for approximately 170 000 tracks and 60 000 users. We pre-process the data by filtering out the tracks and users with too few in-

[2] `jamendo.com`

teractions (less than 5) and the top outliers, resulting in approximately 31 000 tracks and 27 000 users.

We do a pre-analysis of the data to determine the number of factors to be used for the matrix factorization. We use the alternating least squares (ALS) algorithm (Hu et al., 2008b) which is one of the SOTA matrix factorization algorithms.[3] Using a stratified split with a test ratio of 0.2, we evaluate different numbers of factors in terms of the performance using normalized discounted cumulative gain (NDCG) and mean average precision (MAP). The results are shown in Figure 4.1 with 96 factors providing the highest overall performance. We also consider 64 and 128 factors to compare the consistency of several CF spaces.

### 4.4.2 Content-based features

To extract content-based features we use the Essentia library (Bogdanov et al., 2013) and the following music auto-tagging models (Alonso-Jiménez et al., 2020): MusiCNN, VGG and VGGish (see Chapter 3 for more details on these architectures).

The models provided were pre-trained on several datasets. MusiCNN and VGG have been trained on top 50 tags from Million Song Dataset (MSD) (Bertin-Mahieux et al., 2011) and MagnaTagATune (MTAT) (Law et al., 2009). These datasets have been introduced in Chapter 4.4 and contain music and are focused on the music auto-tagging. VGGish has been trained on AudioSet (Gemmeke et al., 2017) which is an audio event recognition dataset that also includes music. It allows us to compare different architectures that have been trained on the same dataset and the same architecture trained on different datasets.

For the MusiCNN and VGG architectures, we consider the latent spaces constructed by the output (taggrams) and the penultimate (embeddings) layers. VGGish model only provides embeddings. The number of dimensions for the layers is summarized in Table 4.1. In total, we extract 9 content-based (CB) feature vectors.

[3]Implementation from `github.com/benfred/implicit`

| Dataset | | Architecture | Layer | Dim |
|---|---|---|---|---|
| MSD | × | MusiCNN | Embeddings | 200 |
| | | | Taggrams | 50 |
| MTAT | | VGG | Embeddings | 256 |
| | | | Taggrams | 50 |
| AudioSet | | VGGish | Embeddings | 128 |

Table 4.1: Dimensions of latent spaces

We attempted to process the 31 000 tracks that we obtained from CF data, but due to some tracks being no longer available or corrupted, this number decreased to approximately 29 000.

### 4.4.3 Final dataset

The final large dataset contains 29 275 tracks with successfully extracted CB features. We repeated the matrix factorization on the collaborative filtering data containing only those tracks (29 275 tracks × 27 235 users, 793 963 non-zero values) with the number of factors of 64, 96, and 128 to obtain the CF features. We release this final dataset with 3 CF and 9 CB representations publicly.

We create a smaller subset of the final dataset that is obtained by intersection with MTG-Jamendo (Bogdanov et al., 2019) test set of split-0, which resulted in 1 372 tracks, which is comparable to a small music collection. We present the experiments on this small dataset, as it visualizes the relative differences between spaces better. [4]

[4]The results on the large dataset are available on the companion website.

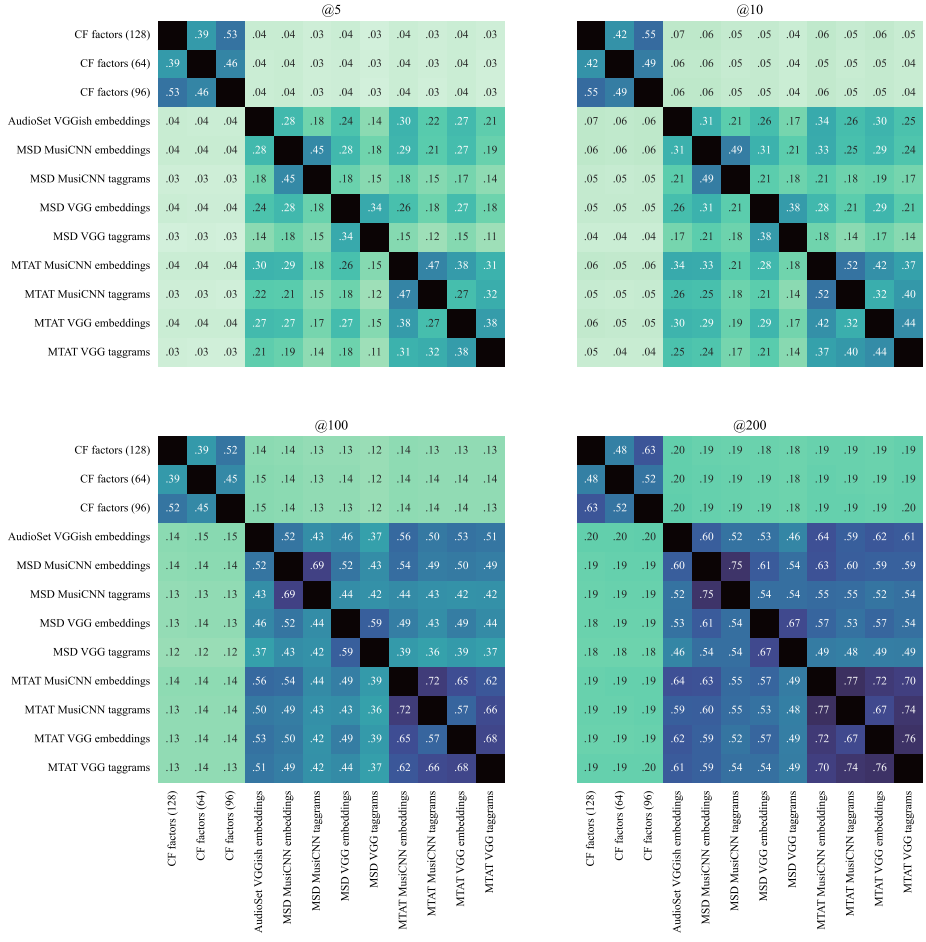



Figure 4.2: Nearest neighbor similarity ($S_n$) of CB vs. CF spaces

## 4.5 Offline Experiments

### 4.5.1 Latent spaces

We compare the collaborative filtering and content-based spaces introduced in Section 4.4 in terms of NN-similarity $S_n$, introduced in Section 4.3. We present the results using cosine distance to calculate nearest neighbors in Figure 4.2. Euclidean[5] and cosine distances produce very similar results, except that NN-similarity between CF rankings using Euclidean distance is slightly lower.

The first thing that stands out in Figure 4.2 is that the CF spaces are quite dissimilar in terms of NN-similarity from CB spaces, as all pairs of rankings that include CF and CB spaces have the lowest values. This indicates that the music similarity captured by CF and CB spaces is noticeably different. The NN-similarity values between CF spaces stays consistent and is among the highest observed overall at all cutoffs. However, there is enough difference between CF spaces (e.g. max $S_5$ is 0.66 which means that 2 out of top-5 tracks will be different) to make the number of CF factors an important design decision.

Related to CB embeddings, there is much more variability in the nearest-neighbors lists at smaller cutoffs. Therefore, the choice of the latent space leads to significantly different outcomes in the use-cases that rely on the small number of nearest neighbors. For example, $S_{10}$ varies between 0.16 to 0.58, which means that 4 to 9 tracks will be different between any two CB spaces. At larger cutoffs (100, 200), the NN-similarity between CB spaces is higher ($S_{200}$ ranges from 0.46 to 0.77 between CB spaces).

We can calculate what choice impacts the NN-similarity more: dataset, architecture, or layer. To analyze this, we can fix the two out of three variables and calculate the average NN-similarity between the pairs that come from comparing the third variable. For example, to determine how much the choice of *dataset* contributes to NN-similarity, we average the $S$ values of MSD vs. MTAT for MusiCNN embeddings, taggrams, VGG embeddings, and taggrams. As we calculate those for a cutoff value of 5, we

[5]More figures available on the companion website.

| Cutoff | 5 | 10 | 100 | 200 |
|---|---|---|---|---|
| Dataset (MSD vs. MTAT) | .26 | .26 | .46 | .56 |
| Arch. (MusiCNN vs. VGG) | .35 | .36 | .57 | .65 |
| Layer (emb. vs. tag.) | .50 | .51 | .68 | .74 |

Table 4.2: Average NN-similarity along the variable

get that the average NN-similarity for choice of the dataset is 0.26, which means that if we change the training dataset, roughly only $0.26 \times 5 \approx 1$ track will be the same in the list of 5 nearest neighbors. The values for all cutoff values are presented in Table 4.2. According to the computed average NN-similarity, latent spaces produced by models trained on different datasets (MSD vs. MTAT) are more dissimilar than those using different architectures (MusiCNN vs. VGG). Indeed MusiCNN and VGG are both CNN-based and share some similarities.

Regarding the choice of the layer (taggrams vs. embeddings), we can observe the highest NN-similarity when comparing spaces generated by the same model (same dataset and architecture). At the same time, taggram spaces are dissimilar to other CB spaces. That makes sense for spaces from different datasets, as the resulting tag spaces have different vocabulary and semantics. Interestingly enough, it also holds for different architectures on MSD (e.g., MSD MusiCNN vs. VGG taggrams $S_5 = 0.18$, which is close to MSD vs. MTAT MusiCNN taggrams: $S_5 = 0.19$). However, the NN-similarity is much higher for MTAT MusiCNN vs. VGG taggrams: $S_5 = 0.40$.

Overall, the MTAT dataset seems to produce spaces that are generally more similar to each other than MSD. It can be attributed to the smaller size of MTAT, where the difference between architectures cannot be as pronounced. However, if the tag predictions produced by different architectures on the same dataset are close, that might indicate the quality of annotations. While MSD annotations come from Last.fm folksonomy (every user can assign any tag), MTAT annotations come from the gamified system, where the annotators are encouraged to assign tags that might

be similar to ones used by the other people (Law et al., 2009).

Another interesting observation is that embeddings of different datasets and architectures, despite having higher dimensionality produce lists of nearest neighbors that are pretty similar to each other (minimum values between CB embedding spaces: $S_5 = 0.30$, $S_{10} = 0.29$, $S_{100} = 0.46$, and $S_{200} = 0.53$). It is especially prominent at lower cutoff values (5, 10).

In an online evaluation with limited resources, selecting a subset of latent spaces may be necessary. Based on the results from Table 4.2, it makes sense to prioritize models trained on the different datasets rather than different architectures.

### 4.5.2 Projections

One of the latent space applications is to visualize the similarity between tracks. Hence, we want to use the same methodology to compare how well the NN-similarity is preserved while being projected on a 2D plane. Although some exploration interfaces use 3D planes, for consistency with previous research on music exploration (Tovstogan et al., 2020), we only consider 2D. We consider PCA (Pearson, 1901), t-SNE (Maaten and Hinton, 2008) and UMAP (McInnes et al., 2018). Moreover, as t-SNE and UMAP are stochastic, we consider two different seeds for each projection to measure the robustness. It does not make sense to compare projections for different datasets or architectures from previously obtained results. However, as embeddings and taggrams are quite similar, we consider both of them. Figure 4.3 shows the results for MSD MusiCNN embeddings and taggrams using Euclidean distance to calculate nearest neighbors, as it firstly makes more sense to use in 2D, and secondly, cosine distance similarity is significantly lower for most pairs.[6]

Comparing different projections, it is evident from Figure 4.3 that t-SNE exhibits the highest NN-similarity to the original spaces. Nevertheless, this projection leads to noticeable changes in rankings (e.g., $S_{10} = 0.42$ for embeddings means that 6 tracks in the top-10 list will be

[6]Figures of cosine distance is available at companion website.

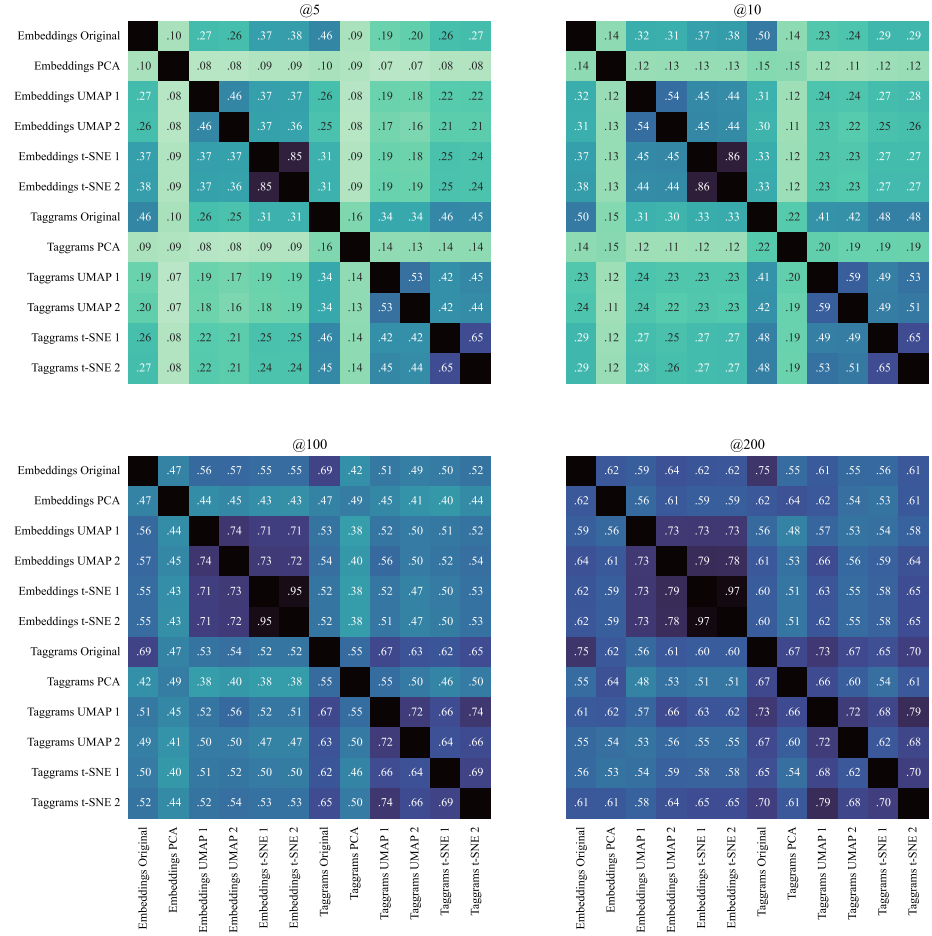



Figure 4.3: Nearest neighbor similarity ($S_n$) of different projections of MSD MusiCNN embeddings and taggrams

different on average). UMAP is the second-best projection, with PCA being the poorest at preserving NN-similarity. The NN-similarity between different seeds for t-SNE is quite close to 1.0, which is also more robust than UMAP. In general, the NN-similarity values are closer to each other at larger cutoff values. As PCA is a linear projection that works well in preserving the global structure of data without much consideration for nearest neighbors, its NN-similarity is relatively low for small cutoff values (5, 10).

From a practical perspective, using t-SNE for projection provides the best results and preserves more than 40% of nearest neighbors for small cutoffs. That means that in the visualization interface, among the five closest tracks in projected space, two tracks are also closest in the original space to the reference track.

## 4.6 Online experiments

Even if music similarity is relatively subjective, it can be partially alleviated by asking more specific questions to the participants, as shown in related work in Section 4.2. We use a methodology similar to (Bogdanov et al., 2009) to evaluate which spaces provide a better representation of music similarity for music recommendation. The difference with other studies is that this methodology evaluates the perception of playlists of top-N similar tracks instead of individual comparisons of pairs of tracks. This approach is better aligned with tasks of music exploration and playlist generation.

We use the same small dataset for this experiment for consistency with offline experiments. For each latent space, the participants are presented with a reference track and several candidate playlists containing the nearest neighbors (ordered by their similarity to the reference). They are asked to rate the similarity of each playlist to the reference track in the hypothetical scenario of music recommendation: *"If you liked how this track sounds, you might like these other tracks"*. The order of reference tracks is the same for all participants, while the playlists are presented in random

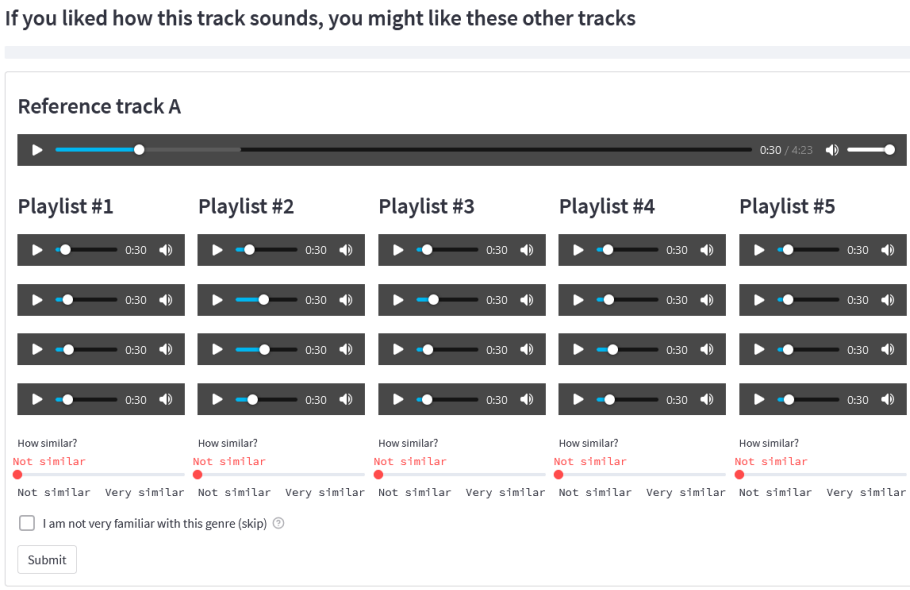



Figure 4.4: Online experiment interface

order.

Because the number of choices presented to participants is limited by possible cognitive overload, we selected the five most dissimilar latent spaces: CF 96, MSD VGG taggrams, MSD MusiCNN taggrams, MTAT MusiCNN embeddings, and VGGish embeddings. We randomly select four reference tracks, ensuring that they are pretty different from each other and span several genres. To keep the time to complete one instance of the experiment as low as possible while providing enough information to the participants, we decided to include four tracks in each playlist, making it 21 tracks per reference track (including the latter) and 84 tracks in total. Because asking participants to listen to each track entirely is unreasonable, by default, we present the participant with a segment of 15 seconds that starts at 0:30 and ends at 0:45. However, the participants can use each player's controls to play more different sections of the track if they feel that they need more information. Participants are encouraged not to spend much time on each track and to use their intuition to rate

the similarity. We communicate that explicitly in the instructions for the experiment. To measure the perceived similarity, we provide a slider that uses a 4-point Likert scale: 0 - not similar, 1 - somewhat similar, 2 - quite similar, and 3 - very similar. We specifically avoided the neutral option to force participants to give their opinion. The interface of the experiment is shown in Figure 4.4.

We provide introductory text that explains the experiment, interface, and purpose and allows the participants to continue with the experiment once they give their explicit consent. After circulating the link to the experiment[7] in the relevant communities (mailing lists, Twitter, subreddits[8]) we obtained data from 39 participants. We asked optional general demographic questions to verify coverage of different demographic groups. All participants are aged from 14–64, with the majority (53%) falling into the age group of 25–34. 55% of participants identify themselves as men, 33% as women, 9% non-binary, and 3% preferred not to say. Concerning music background, 18% do not have any music training, 42% have some, 37% are hobbyists, and 3% (1) are professional musicians. The majority of participants (52%) listen to music on average for 2–3 hours per day, with the whole population listening from less than 1 hour up to 6–7 hours per day.

We use the Shapiro test (p-value$<$0.001) to verify the assumptions for the ANOVA test. We perform ANOVA (p-value$<$0.001) and Kruskal-Wallis (p-value$<$0.001) tests to verify if the choice of the latent space makes the similarity results significantly different. Subsequently, we use Tukey's honestly significantly differenced (HSD) test to identify which pairs of latent spaces are significantly different. The only pair of spaces where the difference is insignificant is AudioSet VGGish vs. MTAT MusiCNN embeddings (p-value of 0.07).

Figure 4.5 shows the average ranking performance of each latent space with the standard deviation represented as a vertical line. We can see as both embedding spaces (AudioSet VGGish, MTAT MusiCNN) that we have chosen for the online experiment perform the best, with no statistical

[7] `philtgun.me/similarity-experiment`
[8] `reddit.com/r/samplesize`

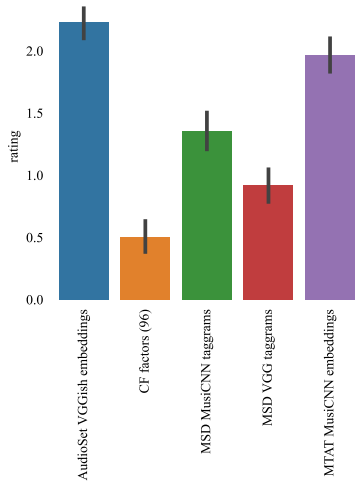



Figure 4.5: Online experiment results

difference between them. An interesting observation is that AudioSet is a generic audio event recognition dataset, and the VGGish model trained on it performs comparably to the embeddings from the music auto-tagging dataset MTAT. MSD MusiCNN taggram space has a worse average similarity rating, with MSD VGG taggrams following it. The poor performance of CF factors space can be attributed to the mismatch of the use-case, as it is intended to be used in conjunction with the user factors, not as a latent space. It is also possible that the small size of the dataset impacted the poor performance of CF factors.

The results show that content-based latent spaces can power the anonymous recommendation systems with a similarity that is rated at least as *quite similar*. It is a positive takeaway for exploration and visualization systems that can be built on top of similar latent spaces.

## 4.7 Conclusions

We compared different collaborative filtering and content-based latent spaces for the nearest-neighbor similarity. We observed that nearest neighbors obtained from CF spaces are very dissimilar to nearest neighbors obtained from CB approaches. Focusing on CB spaces, we identified that the choice of the training dataset (MSD vs. MTAT) tends to produce the most dissimilar spaces, followed by the architecture and then layer. We observed that taggram spaces tend to be dissimilar across different datasets and architecture, while embedding spaces tend to be more similar. Interestingly, the consistency of CB latent spaces derived from a dataset may differ in terms of their nearest-neighbors similarity, as we observed in the MTAT vs. MSD datasets. In the context of 2D visualization of latent spaces, t-SNE exhibits the highest nearest-neighbors similarity between original and projected spaces.

We performed an online experiment to evaluate a selection of dissimilar latent spaces in the context of music similarity for music recommendation. The results show that the CB spaces can be successfully used in music recommendation/exploration scenarios where user-generated data is absent due to design decisions. We observe that embedding spaces (AudioSet VGGish, MTAT MusiCNN) perform significantly better than taggram spaces (MSD MusiCNN, MSD VGG).

Some limitations of our study are that we only work with the embeddings models that are widely known in the MIR research and are reasonably scalable. While there are some newer embedding models proposed recently (Dhariwal et al., 2020), they usually require much more computational power. In practice, it is possible to train many embedding models in the industrial setting.

All analysis[9] and experiment interface[10] code is publicly available on GitHub, under Apache 2.0 license. The latent spaces are published on Zenodo[11] under CC BY-NC-SA 4.0 license, and the audio for the small

[9] `github.com/philtgun/compare-embeddings`
[10] `github.com/philtgun/similarity-experiment`
[11] `doi.org/10.5281/zenodo.6010468`

dataset is available in MTG-Jamendo dataset.[12]

[12] mtg.github.io/mtg-jamendo-dataset

# Chapter 5

# MUSIC EXPLORATION INTERFACE

## 5.1 Introduction

In this chapter, we take advantage of the auto-tagging systems trained to predict the music tags (genre, moods, instruments, etc.) and use the extracted embeddings to visualize music collections. As mentioned in Chapter 3, with the wide usage of deep learning in music information retrieval, the feature extraction moved from careful engineering to learned features. There are multiple pre-trained feature-extractor models available (Hershey et al., 2017; Cramer et al., 2019) that can be used to extract embeddings from audio. Often, these embeddings are used as input for dense neural networks for particular downstream tasks (Alonso-Jiménez et al., 2020). However, they can also be used to represent the music within the embedding space.

We introduce the interface that allows users to visualize their entire collection or subsets of their collection in terms of embeddings extracted from different models and compare them qualitatively. We evaluate the interface in terms of how useful it is for the users to explore their library and create a playlist of the music they have forgotten and would like to rediscover. In addition, we evaluate different models in terms of the users'

preferences for the visualizations that have been produced.

## 5.2 State of the Art

Many research works investigated the visualization of the music in 2D and 3D space for exploration, navigation, and recommendation. Knees et al. (2020) do a comprehensive overview of many of those works and identify 3 phases of music discovery interfaces:

1. *Content-based music retrieval interfaces*: audio processing features and intention of grouping similarly-sounding music together, primarily for small-scale music collections.

2. *Collaborative and Automatic Semantic Description*: interfaces that attempt to aggregate the collaborative user-generated data from online platforms

3. *Recommender Interfaces and Continuous Streaming*: interfaces integrated with streaming services, using data available online

We will take a look at the interfaces that are important for the context of this thesis, and we will identify several aspects that are important for us:

- Features used for visualization
- Dimensionality reduction techniques
- Presence of user studies to evaluate the interface

The summary is presented in Table 5.1 with the rest of this section providing more details about those works.

One of the earliest works is *GenreSpace* (Tzanetakis, 2001) visualizes tracks in 3D space with colors representing genres. As the core contribution of the Tzanetakis was the methodology for automatic genre classification, and the interface was one of the applications, there was no user evaluation associated.

| Name | D | Features | Dims | Reduction | Pers.[a] | Users |
|---|---|---|---|---|---|---|
| *GenreSpace* | 3D | Timbre, rhythm | 17 | PCA | No | No |
| *Islands of Music* | 2D | Rhythm patt. | 1200 | PCA, SOM | No | No |
| ***NepTune*** | 3D | Rhythm FP | 1200 | SOM | Yes | 8 |
| *Globe of Music* | 3D | Spectrum feat. | 168 | GeoSOM | No | 12 |
| *MusicMiner* | 2D | Top low-level | 20 | ESOM | No | No |
| *MYMO* | - | Reviews (NLP) | 3313 | SOM | No | No |
| *SongExplorer* | 2D | Emotions prob. | 7 | SOM | No | Yes (?) |
| *Artist Map* | 2D | Metadata | 4 | FDG[b] | No | No |
| Torrens et al. | 2D | Metadata | 5 | Heur. | Yes | No |
| *Musicream* | 2D | Timbre, genre | 30 | - | No | 27 |
| *MusicGalaxy* | 2D | Timbre, rhythm+ | 4 | LMDS | No | 112 |
| **Vad et al.** | 2D | Mood prob. | 30+ | t-SNE | Yes | 8 |
| *MoodPlay* | 2D | Mood vector | 289 | Corr. An.[c] | No | 279 |
| *Songrium* | 2D | Latent repr. | 200 | PCA | No | No |
| *InstruDive* | 2D | Instr. prob. | 11 | t-SNE | No | No |

Table 5.1: SOTA of interfaces for music visualization
Studies that consider both personal collections and perform user-centric evaluation are marked in bold

[a] Deals with personal music collections
[b] Force-directed graph
[c] Correspondence analysis

### 5.2.1 SOM-based interfaces

One of the most famous interfaces is *Islands of Music* (Pampalk et al., 2002) that uses a self-organizing map (SOM) (Kohonen, 2001) for visualizing music as an artificial landscape of the islands (dense clusters) in the ocean (sparse regions). The emerging islands roughly correspond to the genres of music, and the evaluation is performed primarily qualitatively by authors. The extension of the work (Pampalk et al., 2004) introduces several views (based on timbre, rhythm, metadata features) and the ability to switch between them. Moreover, there was also another related work by Neumayer et al. (2005) that proposed methodology for playlist generation by drawing the trajectory on the map.

In the following years, multiple studies were published that also used SOM or some variation of it. *NepTune* (Knees et al., 2006), inspired by Islands of Music, visualizes the space as a terrain that can be navigated in 3D by a user. The interface was exhibited in public, where the users could explore their collections. The authors conducted a small informal user study (8 participants) to ask for opinions about the interface, which were reported to be very positive.

*Globe of Music* (Leitich and Topf, 2007) projects the space onto sphere instead a plane with the use of GeoSOM (Wu and Takatsuka, 2006). The authors evaluated their system in the form of a questionnaire combined with a semi-structured interview with 12 users. The focus was on the interaction with the system, particularly on the given music collection characteristics and ease of navigation of the system.

*MusicMiner* (Mörchen et al., 2005) uses emerging SOM (ESOM) (Ultsch, 1992) and U-Map to visualize transitions between genre-based groups. The authors used a novel separation metric based on Pareto Density Estimation (PDE) to select the best 20 from over 400 low-level features that could separate individual genres of music (e.g., electronic vs. all other). Moreover, the authors thoroughly compared their selection of the features against other typically used ones in terms of "clustering and visualizing different sounding music". Their system was released as software, but no user evaluation was performed in the paper.

Vembu and Baumann (2004) use SOM together with natural language processing (NLP) of the Amazon artist reviews. The authors performed an offline evaluation in terms of the results of the external recommendation service matching the distribution on their SOM. However, the paper only discusses the possible integration of SOM-based artist similarity into the proposed interface (*MYMO*) without implementation or evaluation.

*SongExplorer* (Julià and Jordà, 2009) is a tangible tabletop interface that presents the songs in a hexagonal grid, also using SOM to project 7-dimensional emotion feature space to 2D. The interface was evaluated by presenting it to users and giving the task to "find something interesting" in the collection. The evaluation questionnaire focuses on "subjective experience, adequacy of the visualization and the organization, and interaction" with positive feedback in all three areas.

### 5.2.2 Non-SOM-based interfaces

Some interfaces use the metadata in various creative ways for visualization. *Artist Map* (Gulik and Vignoli, 2005) visualizes artists based on the metadata (release year, tempo) in conjunction with low-level audio features such as tempo and high-level ones, such as genres and moods. Authors develop the visualization algorithm that uses music similarity for clustering and metadata "magnets" to provide semantics to dimensions. One of the use-cases of the interface is playlist creation, which can be achieved by drawing regions or paths. Sadly, the user studies are only mentioned as future work.

Torrens et al. (2004) focus on visualizing personal music collections in the form of a disc, rectangle, or tree-map organized according to metadata (genre, sub-genre, year, artist) and highlighted according to personal ratings or playcount with the ability to highlight playlists. The authors have performed no user studies.

*Musicream* (Goto and Goto, 2009) is an interface that does not visualize the whole collection at once but presents the user with the flow of the disks. The users are encouraged to actively interact with the interface and listen to tracks presented. Once they find something interesting, they can

save the track and get more similar tracks in the flow. Authors use similarity vectors from Tzanetakis and Cook (2002), but as the paper states, any similarity measure can be used. The authors performed a user study with 27 participants to evaluate the interface with mostly positive feedback.

*MusicGalaxy* (Stober and Nürnberger, 2010) uses multiple so-called facets (timbre: GMMs of MFCCs, rhythm: FPs, dynamics, and lyrics: tf-IDF) to compute similarity for the visualization. Each facet is used to calculate distances independently of others. One of the important features of this paper is an adaptive non-linear multi-focus zoom lens that alleviates the impact of projection distortions. The authors also spend much effort optimizing the implementation to achieve real-time responsiveness. The authors also performed an extensive user evaluation of the prototype (presented at the fair, feedback was collected from 112 visitors). Several conclusions include that projection-based visualization is preferred to the list views, and younger users welcome interactivity. Three testers tested the second prototype with eye-tracking, including various usability improvements.

Since 2010 and the emergence of music streaming, studies have started to focus more on web audio and digital collections. A probabilistic projection of personal music collections based on moods (Vad et al., 2015) is a remarkable study that focused a lot on user evaluation. It uses the mood features that were extracted via the MoodAgent[1] commercial service from personal Spotify libraries. The features are projected with t-SNE (Maaten and Hinton, 2008), and the interface includes background highlighting based on the probabilistic models to show moods with different colors. The system enables playlist generation via both region selection and drawing trajectories. The authors performed a user study with eight participants over two weeks with overall positive responses and multiple valuable insights that include a preference for region selection over trajectory drawing. The authors have also mentioned the concept of rediscovery in this work.

*MoodPlay* (Andjelkovic et al., 2019) is another remarkable 2D interface that visualizes artists on a mood space. The authors also used

[1] `moodagent.com`

a commercial service (Rovi) to obtain mood descriptors for the artists and performed correspondence analysis to calculate similarity. While the free-form exploration is supported, the system is presented as a recommendation system that recommends the artists based on moods. The authors conducted a pervasive user study from human-computer interaction (HCI) perspective that provides multiple insights. The participants were recruited in the Mechanical Turk[2] platform. There is an online implementation of the interface available[3].

To mention several more recent interesting works, *Songrium* (Hamasaki et al., 2015) is a comprehensive interface that uses learned music similarity from audio features to visualize web-native audio in both 3D and 2D. The platform has much functionality to visualize derivative works, chronological sequences, and play music. Songrium was introduced in 2012 and had over 147 000 website visitors. However, no user evaluation was mentioned in the paper.

*InstruDive* (Takahashi et al., 2018) presents the visualization of tracks according to instrument classification. The mapping is achieved t-SNE of absolute appearance rate of 11-dimensional instrument space in scattering mode. The authors also provide a circular visualization based on the shortest route from the traveling salesman problem. No user evaluation is mentioned in the paper.

### 5.2.3 Summary

One common thing in all these works is that the visualization unit is either a music track, artist, or album. As music similarity is a well-researched area of MIR, and it was a task in MIREX until 2014, the similarity on the level of tracks can go only so far until the subjectivity gets in the way (Flexer, 2014). Our approach is to work with the segments of the music tracks on a smaller scale, which might alleviate the subjectivity of the similarity.

[2] `www.mturk.com`

[3] `moodplay.pythonanywhere.com`

Moreover, only several of the mentioned interfaces (Knees et al., 2006; Torrens et al., 2004; Vad et al., 2015) work with the personal music collections. Our study focuses on the rediscovery of personal music collections and directly works with the audio files without external commercial service. Also, most of the works, save for a few exceptions (Mörchen et al., 2005; Hamasaki et al., 2015; Andjelkovic et al., 2019) have never been released publicly, as they have been used for study as prototypes. Furthermore, there is a lack of music exploration systems that use the latest state-of-the-art MIR, particularly deep embeddings.

Another common issue with the related work is that many of the mentioned papers (except for a few notable exceptions) do not perform conclusive user evaluation, which is important for user-centric MIR systems (Schedl and Flexer, 2012). However, several of the above-mentioned interfaces perform exhaustive user studies (Vad et al., 2015; Andjelkovic et al., 2019; Stober and Nürnberger, 2010). The studies that do both are marked in bold in Table 5.1. We conduct a user study to evaluate our system through semi-structured user interviews to get feedback and analyze the functionality in the context of rediscovery and exploration of the personal music collections.

## 5.3 Models

We use Essentia[4] library (Bogdanov et al., 2013) to process audio and extract representations. We use the audio embeddings extracted with modern deep auto-tagging models to represent music in the embedding space and distances between embeddings as a measure of similarity (which has been used for the music recommendation in Ferraro et al. (2021a)).

We use the same architectures from Chapter 4: MusiCNN (Pons and Serra, 2019) and VGG (Choi et al., 2016) (see Section 3 for more details) pre-trained on Million Song Dataset (MSD) (Bertin-Mahieux et al., 2011) and MagnaTagATune (MTAT) (Law et al., 2009) (see Section 2), and VGGish Hershey et al. (2017) pre-trained on AudioSet Gemmeke

[4] `essentia.upf.edu`

et al. (2017).

While MTAT is significantly smaller and usually training on larger datasets gives higher accuracy on downstream tasks, the labels are less noisy, and it provides an excellent second option for the system. As we mentioned in Chapter 4, different MTAT embeddings and taggrams spaces are more similar to each other in terms of retrieved lists of nearest neighbors than MSD. The top 50 most frequent tags from each dataset were used for training the models.

We use the same two layers as in experiments of Chapter 4 in the models' outputs to generate the visualizations in our system:

- *Taggrams* - the output layer that provides tag activation values. The dimension of this layer is 50 for all our models, as they have been trained on top 50 tags.

- *Embeddings* - the penultimate layer of the model. The dimension of embeddings is 200 for MusiCNN and 256 for VGG (summarized in Table 4.1).

We process the audio with the hop size equal to the receptive field of the model (3 seconds), which means no overlapping of the frames. We call the part of the audio of the size of the receptive field that produces one vector of output values a *segment*. Thus, the track is represented by a two-dimensional array with a vertical dimension equal to the extracted layer dimension and the horizontal (time) dimension equal to the track's duration divided by the size of the model receptive field.

## 5.4 Implementation

The system is implemented in Python as a Flask web app. The code is open-source and available on GitHub[5] under GNU Affero General Public License v3.0. The rest of this section will provide details of the data processing pipeline.

[5] `github.com/MTG/music-explore`

First, the audio is indexed in the newly created local SQL database[6] with the track, artist, album, and genre metadata imported from ID3 tags. Next, the audio is processed with the Essentia library with the output of several layers. The advantage of using Essentia is that the models are easy to use out-of-box and that if one has a working CUDA installation, it will be used to do TensorFlow inferencing. We extract both the tag activation values (taggrams) and the activations from the penultimate layer (embeddings). The taggram and embedding vectors are stacked for every audio segment, resulting in a two-dimensional representation of the track, which is saved as a .npy file.

After the data for all tracks have been extracted, the PCA (Pearson, 1901) projection of the embeddings and taggrams is performed. We also compute STD-PCA projection for the second iteration, where each embedding/taggram vertical dimension is first normalized on the whole population to prevent large variation ranges in the activation values from dominating the PCA-projected space. For retrieval efficiency, the taggrams and embeddings are then aggregated into one .npy file per model. The segments are indexed in the database for easy lookup of the associated track.

We use Plotly[7] library to visualize the embeddings. It is a robust library that works well for our use case. One of its advantages is that it supports multiple programming languages, so it is possible to generate plots in Python and add the interactivity in JavaScript.

## 5.5 First iteration of interface

The first iteration of the interface (Tovstogan et al., 2020) (Figure 5.1) was built with the primary motivation of quick qualitative evaluation of the auto-tagging models. We use the models mentioned in Section 5.3 to extract embeddings for the MTG-Jamendo dataset. The choice of MTG-Jamendo dataset allows us not to host audio and use Jamendo API to serve

[6] SQLite via SQLAlchemy

[7] `plotly.com`

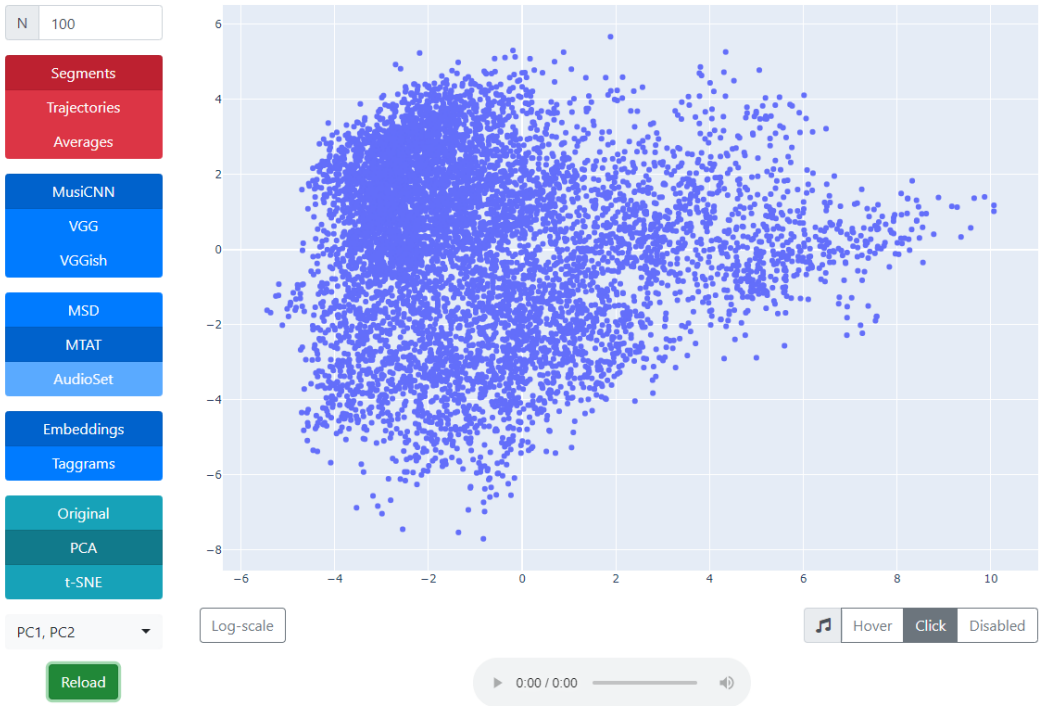



Figure 5.1: System interface (first iteration)

the audio.

The data controls are located on the left side of the interface. The first field is used to limit the number of tracks that are visualized, as too much data visualized simultaneously affects the performance of the interface and clarity of visualization. Depending on the system's technical capabilities, it can vary: 100 tracks can be handled with no problem on most systems (MacBook Air 2017), while more powerful ones could handle 300–500.

Colored in the red there are three modes to visualize tracks:

- *Segments* – each segment is visualized as a separate point on the graph (Figure 5.1).

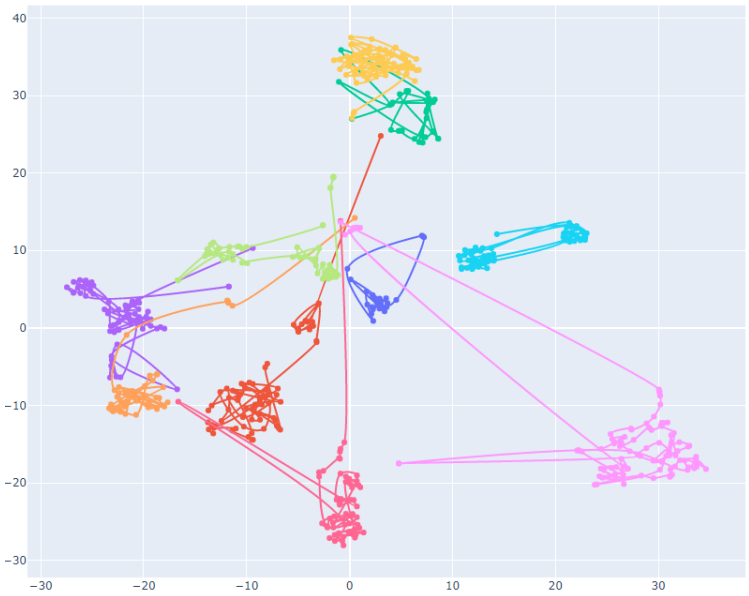
- *Trajectories* – each track is visualized as a line that connects its consecutive segments (Figure 5.2a). It allows for the visualization of separate tracks but works well only for a small number of tracks (10 or less).

- *Averages* – The track is represented as an arithmetic mean of the values of its segments, visualized as a circle with a diameter proportional to the standard deviation (Figure 5.2b).

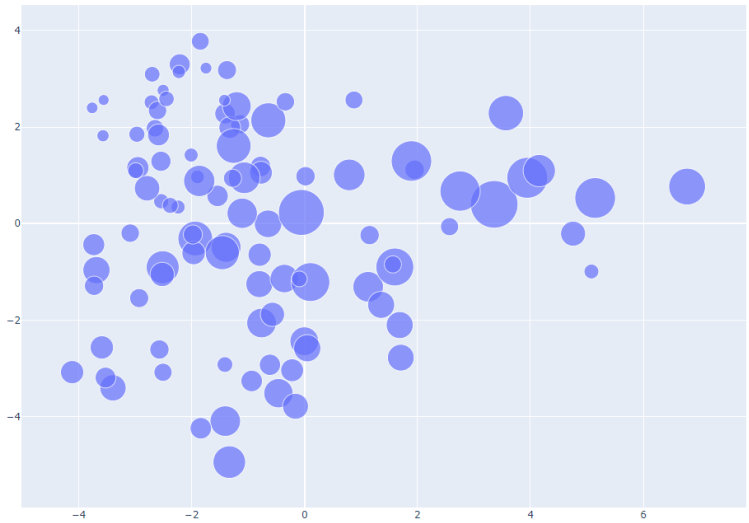
One can listen to that particular segment or track by hovering or clicking on the point. It is also possible to zoom in and out to look at clusters of points or areas on the graph that might be of particular interest.

Colored in blue are the controls to select the architecture and training dataset and which layer to visualize. Colored in teal are the projection options that can be used for dimensionality reduction: PCA (Pearson, 1901), and t-SNE (Maaten and Hinton, 2008). It is also possible to visualize the original dimensions. In the case of taggrams, they directly correspond to the tag that the model is predicting. In the case of embeddings, looking at original dimensions is not very useful; thus, dimensionality reduction techniques are useful here.

Visualizing individual tags in taggrams is very useful for quick qualitative evaluation of the auto-tagging system. By listening to segments




(a) Trajectories




(b) Averages

Figure 5.2: Viewing modes

with high activation values, one can immediately hear if the tag is representative (e.g., hearing *guitar* or comparing if it is *slow*er than other segments). Also, in the case of noisy labels it easy to see if the tags with the same semantic meaning have a high correlation (e.g. *vocal* and *vocals* in MTAT), or that semantically mutually exclusive tags have negative correlation (e.g. *vocal* and *no vocals* in MTAT).

The interface is also helpful for exploring music. By selecting tags to look for new music at the intersection of genres/categories or using dimensionality reduction, one can explore the whole latent space from different perspectives. Distance between the points is indicative of similarity relative to selected tags.

Dimensionality reduction also allows for exploring the semantics of the learned embedding space. Listening to segments while slowly moving along one of the axes can give insight into the semantics of the most significant differences learned by auto-tagging systems.

Looking at trajectories with t-SNE can give insights into the structure of the tracks and their temporal evolution. For example, transitions between vocal and instrumental parts of the track are pretty evident (two cyan clusters in Figure 5.2a).

## 5.6 Second iteration of interface

The second iteration of the interface was designed to address some of the comments from the first paper. We have a working prototype (Figure 5.3) that we evaluate with users to compare visualization; therefore, it has two panes. In a final system, it can be reduced to one pane only.

The interface is split into several sections: music selection, visualization selection, and highlighting. The user can select music to visualize by selecting the tags of interest or artists. One of the essential aspects of the system is that it does not average the individual embeddings of the song segments. Each segment is of the appropriate length of the input size of the model (3 sec for both MusiCNN and VGG architectures).

One point on the graph represents one segment. The reduction slider


Figure 5.3: System interface (second iteration)

allows showing fewer segments per track to visualize many tracks at once. The number represents the step size when loading the data, so it shows all segments for a value of 1, skips every other segment for the value of 2, skips two for the value of 3, etc.

The highlighting section allows highlighting one or more artists, tags, albums, or tracks in red color on the graph. It is interesting to see the groupings and spread of the particular subset of the collection in the context of the more extensive selection of music.

The visualization selection controls above the graphs allow a user to select architecture, dataset, layer, and projection to visualize embeddings. The option names have been anonymized during the user study to remove any bias the participants might have towards any options. Each option can be selected individually to facilitate the comparison of the combinations. For example, the user might only change the dataset while keeping all other fields the same to see how the training dataset impacts the embedding space visualization.

The available architectures, datasets, and layers have been described in Section 5.3. Among the available projections, apart from PCA and t-SNE, we also introduced STD-PCA and UMAP (McInnes et al., 2018). PCA and STD-PCA are computed after the extraction of the embeddings. T-SNE and UMAP are computed dynamically upon user request. So while they are slower initially, a caching layer is implemented to prevent repeated computation of the projections of the same subset.

To get an impression of how different the embeddings spaces are, Figure 5.4 shows one of the users' personal music collections that was used for evaluation (with a reduction value of 20). This collection mainly consists of rock and metal music. Highlighted in red is the artist *Enigma* which is tagged as *new age*. While it is mostly condensed in one part of the visualizations, some architecture/dataset/layer combinations manage to cluster it better.

There are several features of the system to facilitate interactivity. The user can listen to the music while hovering or by clicking the point on the graph representing a track segment. Moreover, when the label of the segment is displayed on one graph, the same label for the same segment is

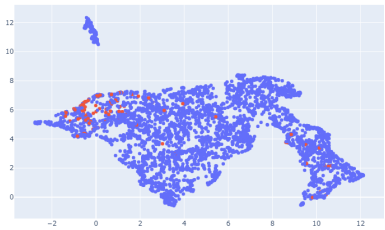
displayed on another graph (see Figure 5.3). It enables easy identification of the same segment on both graphs during an interaction. Moreover, the user can select several segments on one graph with the lasso or box selection, and the corresponding segments will also be selected on the second graph. More tools are available to zoom in and pan the individual graphs to delve deeper into exploring the cluster of interest.

## 5.7 Experiments

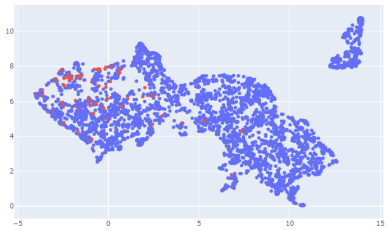
We invited eight users with personal music collections to participate in a user study by authors' colleagues to test the system. We conducted individual semi-structured interviews with each participant to gather feedback and assess the usability and viability of the system. While there are a lot of potential uses for the system, we focus on the use case of exploration and rediscovery of the music in the private personal collections. We want to address two main research questions: the system's feasibility for the exploration and rediscovery of the users' music collection and the comparison of visualizations in terms of usefulness and interest to users.

While we could potentially use the MTG-Jamendo dataset for the evaluation, we decided to use users' personal music collections. Firstly, participants are familiar with their catalog (even for rediscovery) and will be knowledgeable in their judgments. Another reason for familiarity is for participants to engage with the system personally. Secondly, it is helpful to see how the system performs for different collections of different music and reduce the dataset bias for the evaluation.

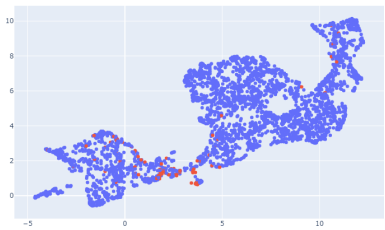
Before the experiment, the participants were asked to select a subset of their private music collection that they wanted to explore. We recommended the participants limit the subset to no more than 1,000 tracks, and in practice, we encountered collections of sizes from 400 to 1,200. In the remote setup, we communicated with the participants through chat to help with data extraction and ensure that the system ran on the users' machines. Then we conducted a video conferencing call with the participant sharing their screen. We asked participants to bring the music collection



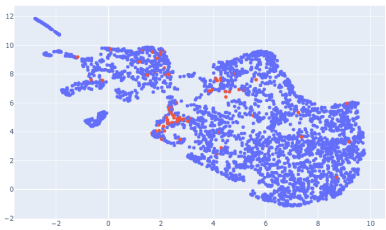
(a) VGG MSD taggrams



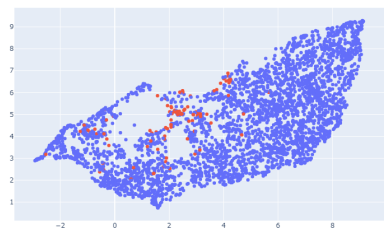
(b) MusiCNN MSD taggrams



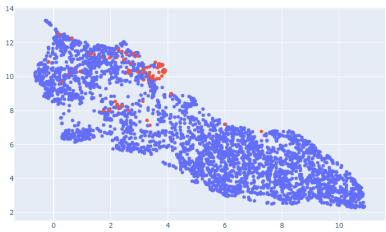
(c) VGG MTT taggrams



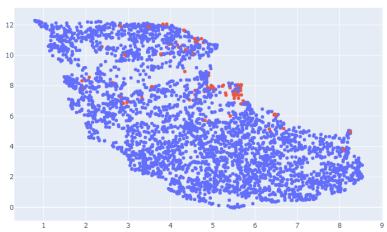
(d) MusiCNN MTT taggrams



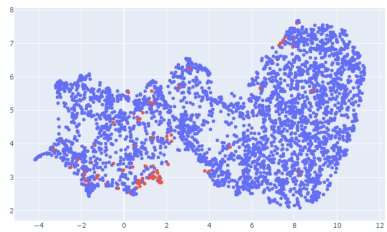
(e) VGG MSD embeddings



(f) MusiCNN MSD embeddings



(g) VGG MTT embeddings



(h) MusiCNN MTT embeddings

Figure 5.4: UMAP visualizations of *new age* (in red) in mostly *rock* and *metal* collection (reduction of 20)

on the external storage device and performed data extraction and setup on the authors' machines in the live setup. Of 8 participants, one was interviewed remotely, and 7 — in-person. The data extraction took different times depending on the specification of the user machine: from 1 to 4 hours with an average of 1.5 hours. While the system does not require GPU for processing, most participants used machines with CUDA installation, which sped up the extraction process drastically.

The video and audio from the call and audio from the live interview were recorded with the participant's consent for further transcript analysis. The experiment started with introducing the features of the system to the participants by reading the introduction text. The text was kept the same to minimize the possible bias. We let participants get familiar, ask questions, play with the system, and make sure that they are comfortable with it. The maximum time allocated for the familiarity phase is 10 minutes. We ensured that participants used every part of the interface at least two times, and if they did not, we encouraged them to use it. Then we gave the participants a task that was formulated as such: *"Imagine that you want to listen to something from your library that you have not listened to in a while. Explore the system and make a playlist for yourself."*

During the interview, the participants were encouraged to try different settings and engage with the system as much as possible. When they changed the visualization parameters (architecture, dataset, layer, and projection), we asked them if they liked or disliked the previous combination. After the users were content with their selection of the tracks for the playlist, we asked them to fill in the questionnaire[8] to assess their thoughts about the system.

The questionnaire is split into two parts: the first part included background questions such as age, musical training, familiarity with playlists, and experience with listening to music. The authors were present to answer any questions the users might have about the questions but did not interfere beyond that.

The second part of the questionnaire contains questions about the system designed to identify which features the system users like, what they

[8]The questions are available in the Appendix C

thought about the visualizations on both macro and micro levels, the system's usefulness for music exploration, rediscovery, and playlist creation. To measure users' opinions and feedback, we used the 5-point Likert scale: 1 - Strongly disagree, 2 - Disagree, 3 - Neither agree nor disagree, 4 - Agree, 5 - Strongly agree. Interviewees were asked to be as critical as possible and encouraged to explain their reasoning behind the choices they made and think out loud.

## 5.8 Results and Discussion

The participants of our study are aged 27-39 years with an average age of 30, 7 male and 1 female. They all have some music training ranging from 1 to 20 years, the median of 6, and an average of 8 years. They listen to music for 0.5-8 hours per day with 1 hour or less actively, less than 50%, 20% on average to playlists. The participants create playlists with frequency ranging from every day to almost never, with good coverage of all options. The frequency of desire to rediscover their music ranges from every day to several times per month, with most of the answers in the latter category. The broad genres covered by the users' personal music collections span mainly electronic, rock and metal.

### 5.8.1 Interaction, exploration and rediscovery

After analyzing the interviews and the results of the survey (see Table 5.2), we can see the trend that the system achieves its goal of helping users to interact, explore and rediscover personal music collections and create playlists. The feedback is very positive, with every participant having discovered some exciting connections between tracks in their library during the interviews.

One of the topics that came up in several interviews was about using *segments* instead of tracks, segment length, and possible averaging of the segments. An argument in favor of using segments is that they are short, concise, can represent better the music evolution with time and span mul-

| Question | Mean $\pm$ STD |
|---|---|
| Liked interacting with system | $\mathbf{4.9} \pm 0.4$ |
| Had preference for particular model | $3.6 \pm \mathbf{1.2}$ |
| Preferred over browsing | $4.3 \pm 0.7$ |
| Preferred over random | $4.4 \pm 0.9$ |
| Liked big picture | $3.8 \pm 1.0$ |
| Liked segment groupings | $4.4 \pm 0.7$ |
| *Discovered unexpected connections* | $\mathbf{4.5} \pm 0.5$ |
| *Rediscovered something* | $\mathbf{4.6} \pm \mathbf{1.1}$ |
| Want to use for playlist creation | $4.1 \pm 1.0$ |
| Want to use for inspiration | $4.3 \pm 0.7$ |
| Had rewarding experience | $4.1 \pm \mathbf{1.1}$ |
| Had engaging experience | $\mathbf{4.5} \pm 0.8$ |

Table 5.2: Summarized results from Likert scale questions

tiple tags, and are easier to perceive as a unit. For example, while it might be challenging to say which track is more similar to the reference track, some participants agreed that it is relatively easier to answer the same question with the segments.

However, multiple participants remarked that the length of 3 seconds was too short. While the similarity might be easier to judge, it might not translate well towards track similarity and exploration process and lead to undesirable behavior during playlist generation. For example, if there is a segment of low-energy music in the cluster of similarly chill tracks, but the segment is an interlude in a much more aggressive track, the track in question will be undesirable in the low-energy playlist. Another issue with the 3-second segments that was raised is that the duration is too short for exploration and rediscovery. Some participants mentioned that they would prefer segments of at least 10 seconds.

One suggestion that came up multiple times is to *average* embeddings of several segments. It makes sense for the segments that are similar to each other. However, if the segments are pretty distinct and are from two

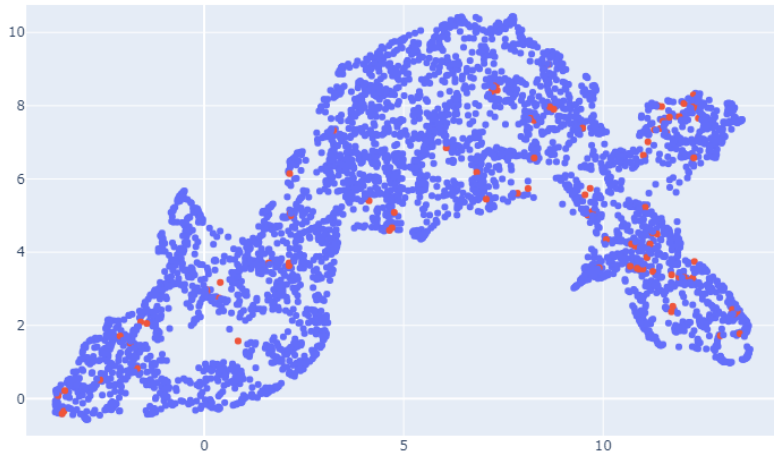



Figure 5.5: Long complex track highlighted in red

different regions in the embedding space, taking the average might put the resulting average into a new third region with nothing to do with the original ones. This problem is exacerbated on a larger scale, where averaging can make tracks that are very complex and span multiple regions in the embedding space (Figure 5.5) be reduced to several points which are not representative of the dynamics of the track.

Participants' opinions varied regarding the ability of the interface to visualize the entire collection. One of the participants noted that it was nice that all aggressive and high-energy tracks were on one side with the more chill and relaxed tracks on the other side (PCA). Other participants just enjoyed hovering the mouse over the different regions of embedding space without trying to make sense of the global distribution. Some participants enjoyed zooming into random clusters and exploring them without much interaction on the global scale.

Participants' opinions varied regarding the ability of the interface to *visualize the entire collection*. Some participants noted that it was nice that all aggressive and high-energy tracks were on one side with the more chill and relaxed tracks on the other side. One participant mentioned

*"(pointing at one side of the visualization) here is hard music, music that my mother does not like, but if I come here (pointing at the opposite side), it is more peaceful, relaxing"* while moving from one side of the visualization to the opposite one. The semantics gradients mentioned as evident from the big picture are (depending on the architecture/dataset): rock–ambient, electronic–acoustic, vocal–instrumental. The tags from the training datasets represent those semantics, and it is helpful to see that the participants agree on those semantics. Several other participants did not pay any attention to the global distribution and dived right into exploring clusters hovering the mouse over the different regions of embedding space. Some participants enjoyed zooming into random clusters, while others did not utilize zoom functionality as much.

*Rediscovery* was the part of the experience that almost all the participants were pleased with and vocal. Ones that were not particularly keen on rediscovery evaluated the system more in the context of DJing. Encountering artists and tracks that they have not listened to in a while happened both during the random walks over the entire space and while investigating local clusters. The same can be said about unexpected connections, with several participants saying *"I would never think to put these two artists together in a playlist, but it works quite well for these tracks,"* or *"if you listen to segments, they sound quite similar in timbre, what will not happen to full tracks."* Some participants have noted that it was good to have an audio player in the interface because if they were using the system outside of the interview, they would stop the exploration process and listen to the track that they stumbled upon from start to finish.

Interestingly enough, the *highlighting* functionality of a particular artist / album / track / tag became quite divisive — many participants used it to highlight a tag or an artist either as a seed to go from or as a target that they wanted to explore. This functionality was most often mentioned as a favorite in the questionnaire. However, some participants did not engage with it after the introduction.

As the tags that the models are trained on are pretty generic (guitar, vocal, rock, chill, electronic, etc.), several participants mentioned that the models probably are not capable of distinguishing subtle differences

between sub-genres of their homogeneous collection by pointing out the segments in some clusters that do not belong together due to stylistic. One participant noted: *"The similarity is not captured well within dance music."*

Overall, the participants took between 5 to 10 minutes to get familiar with the system and 2 to 20 minutes to explore it, try different visualizations and make a selection that would produce a playlist that they were satisfied with. However, after they had created the playlist that they were content with, some participants spent much time continuing exploration of other regions of their collection. Several users mentioned that there could be other methods to generate a playlist, for example, track- or artist-based radio that uses the seed segment or track: *"Maybe the system can lasso select tracks for me."* The playlist creation functionality was mentioned multiple times as a solid use case for using the system after the novelty would wear off.

### 5.8.2 Comparison of visualizations

Even if the sample size for the comparison study is not ample to draw firm conclusions, after analyzing the responses to the question of whether the participants liked or disliked a particular combination of architecture/dataset/layer/projection, some interesting insights can be drawn. As mentioned before, all options were anonymized for user testing to remove potential biases. The only option that could be easily inferred was the projection, as participants could guess the type of projection just by looking at the graphs. However, no participants made it evident that they recognized any projections.

Several participants mentioned that they liked two visualizations side-by-side and used both to select subsets. Some participants pointed out that different combinations captured well different aspects of similarity: *"It seems that A2D2 (MusiCNN-MTAT) can separate ambient from drums, while A1D1 (VGG-MSD) gets the timbral aspect of sounds together well"* and took advantage of that by using both at the same time. Multiple participants have mentioned the VGG-MSD combination as being suitable

for timbre similarity.

Among architectures, datasets, layers, and projections, participants had the strongest preferences for projection options. Most participants mentioned that the distribution looks more attractive in UMAP and t-SNE than in PCA and STD-PCA. We attribute it to both t-SNE and UMAP being non-linear transformations and UMAP preserving distances better than t-SNE. Non-linearity helps to represent the local distances better at the cost of the global distribution. The typical comments in favor of PCA and STD-PCA are that they are faster and capture the global picture much better. *"P1 (STD-PCA) seems to group sounds that I would put together for DJing"*

While participants were encouraged to compare different architectures, datasets, and layers, it took much effort and was less engaging than exploring the visualizations already in front of them. We conclude that a separate experiment should present participants with predetermined comparison pairs for proper evaluation. Although, all participants answered positively to the question of them having a favorite combination of architecture / dataset / layer / projection.

Comparing the architectures coupled with datasets, commonly mentioned as good were combinations VGG-MSD ($n = 3$) and VGG-MTT ($n = 3$), a bit less MusiCNN-MTT ($n = 2$). While VGG is an architecture from computer vision that was not modified much, and MusiCNN takes advantage of the music domain knowledge in the filter design, there was no conclusive evidence for one being preferred more than the other. Taggram layer was mentioned several times in the preferred combinations ($n = 4$), more than the embedding layer ($n = 2$). It might indicate that the semantics of the tags is more valuable and representative than the deeper layer of the neural network.

## 5.9 Conclusions

We present the interface that allows users to visualize personal music collections. It is the first study proposing a music exploration interface that

uses state-of-the-art deep audio embeddings to the best of our knowledge. Notably, the system is open-source, the installation process is well documented, and it is easily extendable with other models for extracting feature embeddings.

We evaluated our system via semi-structured interviews with the users. From the evaluation results, we can conclude that this interface is engaging and rewarding to use for people when they are in the mood for rediscovery or exploration of personal music collections. Moreover, the questionnaire results strongly support the usefulness and viability of the system.

While the performed small-scale evaluation provides initial results and insights on the preferences for architectures, training datasets, layers, and projections, a more extensive study need to be conducted to gather more data to support our initial findings. To provide a better comparative analysis of our interface to other methods to create playlists (metadata browsing and random shuffle), it would be useful to implement those as baselines and provide more ways to create playlists.

# Chapter 6

# CONCLUSIONS AND FUTURE WORK

## 6.1 Contributions

In this thesis, we explored the concept of music exploration and rediscovery of personal music collections from the perspective of the deep auto-tagging systems. The contributions of the work are:

- Conducting an anonymous online survey to identify trends in music listening, exploration, and discovery behavior with 330 responses. Considerable evidence shows an opportunity for a better exploration and discovery systems and processes.
- Introduction of a new open dataset for auto-tagging — *MTG-Jamendo* with over 55 thousand creative-commons licensed tracks and 183 tags split over three categories: 87 genres, 40 instruments, and 56 mood/theme. It contains over 500 GB of freely downloadable audio, pre-computed mel-spectrograms, and Essentia features.
- Organization of *Emotion and theme recognition in music using Jamendo* task within the Multimedia Benchmarking Initiative (MediaEval) in 2019–2021. In this way, we promote usage of the MTG-

Jamendo dataset for auto-tagging and provide a framework for researchers to build better auto-tagging systems for moods and themes. After analyzing the results of three years of the task, we identify the approaches that are high-performing or promising.

- Comparison of different collaborative filtering and content-based latent spaces of the state-of-the-art auto-tagging systems regarding the nearest-neighbor similarity. We identified that the choice of the training dataset (MSD vs. MTAT) tends to produce the most dissimilar spaces, followed by the architecture (VGG, MusiCNN), and then layer.

- Performing an online experiment to evaluate a selection of dissimilar latent spaces in the context of music similarity for music recommendation. The results show that the embedding spaces from the penultimate layer perform better than taggrams activations in terms of subjective music similarity in the context of recommendations achieving a rating of at least *quite similar*.

- Building of a web app that allows exploring the MTG-Jamendo dataset from the perspective of state-of-the-art auto-tagging systems. It serves as a proof-of-concept interface that enables quick qualitative evaluation of deep learning architectures.

- Creation of the framework that uses state-of-the-art deep auto-tagging systems to process the personal music collections and visualize them in the web interface providing a multi-faceted novel way for users to explore and rediscover their collections. We have evaluated the interface via semi-structured interviews, with the results confirming the value of such an interface for the rediscovery of personal music collections and playlist creation.

## 6.2 Limitations

One of the most significant limitations is the availability of the music data. Even though we introduced the MTG-Jamendo dataset, the industrial catalogs are more extensive in orders of magnitude. With the larger amount of data, it would be possible to train better models and evaluate them within the MediaEval challenge. Moreover, the quality of the MTG-Jamendo dataset still will not match the quality of catalogs of commercial music.

Another limitation is that we mainly work with the architectures and embeddings widely known in the MIR community in this thesis and do not require a lot of computation power. Many modern architectures are relatively better, but usually, they are more complex and thus require more computational resources and data to train. However, the framework introduced in this thesis can be used in the industrial scenario to extract and evaluate embeddings on a larger scale.

While the evaluation of the interface within this thesis provides exciting and valuable discussion and conclusions, the number of participants can be considered a bit small[1]. While we focus on the qualitative results and insights from the interview, the quantitative results should be interpreted with the awareness of the respective biases.

## 6.3 Open science and reproducibility

We follow the principles of open sciences and make all the code and data produced in this thesis available online to foster reproducibility.

All code related to *MTG-Jamendo* dataset (Chapter 2) is released on the GitHub[2] under Apache 2.0 license. All metadata is available under Creative Commons BY-NC-SA 4.0 license[3]. All audio is available under creative commons licenses. Details for individual track licenses are in the

[1] This thesis was partially conducted during the COVID-19 pandemic

[2] `github.com/MTG/mtg-jamendo-dataset`

[3] `creativecommons.org/licenses/by-nc-sa/4.0`

`audio_licenses.txt` file in the repository. The repository includes all the pre-processing code that was used to generate the final version of the dataset, scripts to download the dataset (audio, spectrograms, pre-computed features), and the baseline PyTorch code. All instructions are included in the `README` file. We mirror the metadata in Zenodo[4], however, as the audio data is too large to be hosted on Zenodo, it is hosted on the MTG servers (Spain) with the mirror on the Google Drive.

While the original baseline code for the baseline from Chapter 2 is part of the MTG-Jamendo repository, the latest version of the baseline is reimplemented with PyTorch Lightning and published on GitHub [5] under Apache 2.0 license. The results reported in the Chapter 2 of this thesis use the seed of 0 and the code from `v0.1.0` tag.

The code for the MediaEval task is part of the MTG-Jamendo dataset, and the code for the task website, as well as the code to process submissions and generate results page, is available on GitHub.[6]. The websites for all editions are hosted with the help of GitHub Pages and are available under their URLs (see Chapter 3). We encouraged all teams to open-source their code, and the individual submissions have links to the team repositories if those were provided.

There are several code repositories for similarity experiments (Chapter 4) that are published on GitHub:

- Analysis and plots[7] (Apache 2.0)
- Interface for online experiment[8] (Apache 2.0)

We also publish the companion website[9] with more figures. The *LatentJam* dataset in published in Zenodo[10] under CC BY-NC-SA 4.0 license.

[4] doi.org/10.5281/zenodo.3826813
[5] github.com/philtgun/mtg-jamendo-baseline
[6] github.com/multimediaeval/2019-Emotion-and-Theme-Recognition-in-Music-Task
[7] github.com/philtgun/compare-embeddings
[8] github.com/philtgun/similarity-experiment
[9] philtgun.me/deep-neighbors
[10] doi.org/10.5281/zenodo.6010468

Code for exploration system (Chapter 5) is published on GitHub[11] under Afero GPL 3.0 License. We provide all the instructions for processing audio and running the system locally in the `README`. There is an online version[12] of the system that allows users to explore the MTG-Jamendo dataset.

## 6.4 Future work

### 6.4.1 Interface

While the visualization of personal music collections can provide users with an engaging and rewarding experience, the research on such interfaces is decades old, and we still do not see any of those in current streaming services. One of the significant issues that we mentioned in Chapter 1 is that exploration and discovery are challenging to quantify. Some metrics involve serendipity, entropy, user engagement, etc. However, the success of music exploration and discovery is only tangible in the long term and needs to be evaluated appropriately. It is not easy to perform long-term user studies, but it is necessary for this type of research.

Although Chapter 5 provides initial findings and insights on the visualization interface, the obvious next step would be a more extensive study with possible integration with popular music streaming platforms, so the participants are not limited to the ones that have digital music collections. However, this research is difficult to conduct without a partnership with the music streaming platform. We successfully collaborated with Jamendo to extract audio features for their music and evaluated them in terms of music similarity, but the integration of the visualization interface is on a much larger scope than a single internship.

[11] `github.com/MTG/music-explore`
[12] `music-explore.upf.edu`

### 6.4.2 MediaEval

After three years of MediaEval task organization, it is evident that the performance of the submitted systems hit a glass ceiling. We have identified that the unevenness of data distribution and a limited amount of data might be impeding the advances of the task. Thus, the extension of this task needs to be investigated, possibly with the introduction of several sub-tasks.

One possibility is to introduce a regression sub-task with predicting arousal and valence, with the ground truth derived from the sentiment analysis of the tags. This sub-task was considered before the first edition of the task in 2019 but was discarded. The reason is the intermediate step to generate the ground truth for arousal and valence values that depend on the quality of the vocabulary mapping.

### 6.4.3 Latent spaces

With the introduction of more systems that are focused on producing the best embeddings either for one specific or multiple MIR tasks (Dhariwal et al., 2020; Castellon et al., 2021; Turian et al., 2022), the proposed framework can take advantage of the better and newer architectures and embeddings for the visualization of music. In the hypothetical scenario, if the streaming services would provide the embeddings via the API, it is possible to adapt the proposed system for visualizing the users' personal libraries or other collections.

## 6.5 Concluding remarks

We hope that this thesis enables and encourages more research on music discovery and exploration and industry adoption of the concepts and interfaces introduced. We stress the importance of user-centric approaches similar to those used in this thesis to evaluate such systems.

While the paradigm of music consumption has changed drastically in

the last decades, the concept of personal collections and music rediscovery should not be ignored by the streaming platforms.

# Bibliography

Aljanaki, A., Yang, Y.-H., and Soleymani, M. (2014). Emotion in music task at MediaEval 2014. In *Working Notes Proceedings of the MediaEval 2014 Workshop*, Barcelona, Spain. CEUR.

Aljanaki, A., Yang, Y.-H., and Soleymani, M. (2015). Emotion in music task at MediaEval 2015. In *Working Notes Proceedings of the MediaEval 2015 Workshop*, Wurzen, Germany. CEUR.

Alonso-Jiménez, P., Bogdanov, D., Pons, J., and Serra, X. (2020). Tensorflow audio models in Essentia. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 266–270, Barcelona, Spain. IEEE.

Amiriparian, S., Gerczuk, M., Coutinho, E., Baird, A., Ottl, S., Milling, M., and Schuller, B. W. (2019). Emotion and themes recognition in music utilising convolutional and recurrent neural networks. In *Working Notes Proceedings of the MediaEval 2019 Workshop*, Sophia Antipolis, France. CEUR.

Andjelkovic, I., Parra, D., and O'Donovan, J. (2019). Moodplay: Interactive music recommendation based on artists' mood similarity. *International Journal of Human-Computer Studies*, 121:142–159.

Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2/3):235–256.

Barraza-Urbina, A. (2017). The exploration-exploitation trade-off in interactive recommender systems. In *Proceedings of the 11th ACM Conference on Recommender Systems (RecSys)*, pages 431–435, Como, Italy. ACM.

Bellogín, A., Cantador, I., and Castells, P. (2010). A study of heterogeneity in recommendations for a social music service. In *Proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec)*, pages 1–8, Barcelona, Spain. ACM.

Bertin-Mahieux, T., Ellis, D. P. W., Whitman, B., and Lamere, P. (2011). The million song dataset. In *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, pages 591–596, Miami, FL, USA. ISMIR.

Bogdanov, D., Serrà, J., Wack, N., and Herrera, P. (2009). From low-level to high-level: Comparative study of music similarity measures. In *2009 11th IEEE International Symposium on Multimedia (ISM)*, pages 453–458, San Diego, CA, USA. IEEE.

Bogdanov, D., Wack, N., Gómez, E., Gulati, S., Herrera, P., Mayor, O., Roma, G., Salamon, J., Zapata, J. R., and Serra, X. (2013). Essentia: An audio analysis library for music information retrieval. In *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR)*, pages 493–498, Curitiba, Brazil. ISMIR.

Bogdanov, D., Won, M., Tovstogan, P., Porter, A., and Serra, X. (2019). The MTG-Jamendo dataset for automatic music tagging. In *Machine Learning for Music Discovery Workshop (ML4MD), International Conference on Machine Learning (ICML)*, Long Beach, CA, USA.

Bour, V. (2021). Frequency dependent convolutions for music tagging. In *Working Notes Proceedings of the MediaEval 2021 Workshop*, Online. CEUR.

Brusilovsky, P., Kobsa, A., and Nejdl, W., editors (2007). *The adaptive web: Methods and strategies of web personalization*, volume 4321. Springer, Berlin, Heidelberg.

Castellon, R., Donahue, C., and Liang, P. (2021). Codified audio language modeling learns useful representations for music information retrieval. In *Proceedings of the 22nd International Society for Music Information Retrieval Conference (ISMIR)*, pages 88–96, Online. ISMIR.

Celma, O. and Herrera, P. (2008). A new approach to evaluating novel recommendations. In *Proceedings of the 2008 ACM Conference on Recommender systems (RecSys)*, pages 179–186, Lausanne, Switzerland. ACM.

Choi, K., Fazekas, G., Cho, K., and Sandler, M. (2018). The effects of noisy labels on deep convolutional neural networks for music tagging. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2(2):139–149.

Choi, K., Fazekas, G., Sandler, M., and Cho, K. (2017). Convolutional recurrent neural networks for music classification. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2392–2396, New Orleans, LA, USA. IEEE.

Choi, K., Fazekas, G., and Sandler, M. B. (2016). Automatic tagging using deep convolutional neural networks. In *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*, pages 805–811, New York City, NY, USA. ISMIR.

Cohen, W. W. and Fan, W. (2000). Web-collaborative filtering: Recommending music by crawling the Web. *Computer Networks*, 33(1-6):685–698.

Cramer, J., Wu, H.-H., Salamon, J., and Bello, J. P. (2019). Look, listen, and learn more: Design choices for deep audio embeddings. In *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3852–3856, Brighton, UK. IEEE.

Cunningham, S. J. and Cunningham, S. J. (2019). Interacting with personal music collections. In *Information in Contemporary Society (ICS)*, pages 526–536, Washington, DC, USA. Springer.

Davis, J. and Goadrich, M. (2006). The relationship between precision-recall and ROC curves. In *Proceedings of the 23rd International Conference on Machine learning (ICML)*, pages 233–240, Pittsburgh, Pennsylvania. ACM.

Defferrard, M., Benzi, K., Vandergheynst, P., and Bresson, X. (2017). FMA: A dataset for music analysis. In *Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR)*, pages 316–323, Suzhou, China. ISMIR.

Dhariwal, P., Jun, H., Payne, C., Kim, J. W., Radford, A., and Sutskever, I. (2020). Jukebox: A generative model for music.

Dipani, A., Iyer, G., and Baths, V. (2020). Recognizing music mood and theme using convolutional neural networks and attention. In *Working Notes Proceedings of the MediaEval 2020 Workshop*, Online. CEUR.

Do, T.-N., Nguyen, M.-T., Nguyen, H.-D., Tran, M.-T., and Cao, X.-N. (2020). HCMUS at MediaEval 2020: Emotion classification using wavenet feature with SpecAugment and EfficientNet. In *Working Notes Proceedings of the MediaEval 2020 Workshop*, Online. CEUR.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874.

Ferraro, A., Favory, X., Drossos, K., Kim, Y., and Bogdanov, D. (2021a). Enriched music representations with multiple cross-modal contrastive learning. *IEEE Signal Processing Letters*, 28:733–737.

Ferraro, A., Kim, Y., Lee, S., Kim, B., Jo, N., Lim, S., Lim, S., Jang, J., Kim, S., Serra, X., and Bogdanov, D. (2021b). Melon playlist dataset: A public dataset for audio-based playlist generation and music tagging.

In *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 536–540, Toronto, ON, Canada. IEEE.

Ferraro, A., Serra, X., and Bauer, C. (2021c). What is fair? Exploring the artists' perspective on the fairness of music streaming platforms. In *18th Human-Computer Interaction Conference (INTERACT)*, volume 12933, pages 562–584, Bari, Italy. Springer.

Flexer, A. (2014). On inter-rater agreement in audio music similarity. In *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR)*, pages 245–250, Taipei, Taiwan. ISMIR.

Flexer, A., Lallai, T., and Rašl, K. (2021). On evaluation of inter- and intra-rater agreement in music recommendation. *Transactions of the International Society for Music Information Retrieval*, 4(1):182.

Flexer, A. and Schnitzer, D. (2009). Album and artist effects for audio similarity at the scale of the web. In *Proceedings of the 6th Sound and Music Computing Conference (SMC)*, pages 59–64, Porto, Portugal. Zenodo.

Foltz, P. W. and Dumais, S. T. (1992). Personalized information delivery: An analysis of information filtering methods. *Communications of the ACM*, 35(12):51–60.

Fu, Z., Lu, G., Ting, K. M., and Zhang, D. (2011). A survey of audio-based music classification and annotation. *IEEE Transactions on Multimedia*, 13(2):303–319.

Gemmeke, J. F., Ellis, D. P. W., Freedman, D., Jansen, A., Lawrence, W., Moore, R. C., Plakal, M., and Ritter, M. (2017). Audio Set: An ontology and human-labeled dataset for audio events. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 776–780, New Orleans, LA, USA. IEEE.

Gerczuk, M., Amiriparian, S., Ottl, S., Rajamani, S. T., and Schuller, B. W. (2020). Emotion and themes recognition in music with convolutional and recurrent attention-blocks. In *Working Notes Proceedings of the MediaEval 2020 Workshop*, Online. CEUR.

Goldberg, D., Nichols, D., Oki, B. M., and Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12):61–70.

Gomez-Canon, J. S., Cano, E., Eerola, T., Herrera, P., Hu, X., Yang, Y.-H., and Gomez, E. (2021). Music emotion recognition: Toward new, robust standards in personalized and context-sensitive applications. *IEEE Signal Processing Magazine*, 38(6):106–114.

Goto, M. and Goto, T. (2009). Musicream: Integrated music-listening interface for active, flexible, and unexpected encounters with musical pieces. *Journal of Information Processing*, 17:292–305.

Gulik, R. v. and Vignoli, F. (2005). Visual playlist generation on the artist map. In *Proceedings of the 6th International Society for Music Information Retrieval Conference (ISMIR)*, pages 520–523, London, UK. ISMIR.

Hamasaki, M., Goto, M., and Nakano, T. (2015). Songrium: Browsing and listening environment for music content creation community. In *Proceedings of the 12th Sound and Music Computing Conference (SMC)*, pages 23–30, Maynooth, Ireland. Zenodo.

Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., and Sugiyama, M. (2018). Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, pages 8536–8546, Montreal, Canada. Curran Associates, Inc.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and*

*Pattern Recognition (CVPR)*, pages 770–778, Las Vegas, NV, USA. IEEE.

Hershey, S., Chaudhuri, S., Ellis, D. P. W., Gemmeke, J. F., Jansen, A., Moore, R. C., Plakal, M., Platt, D., Saurous, R. A., Seybold, B., Slaney, M., Weiss, R. J., and Wilson, K. (2017). CNN architectures for large-scale audio classification. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 131–135, New Orleans, LA, USA. IEEE.

Hu, X., Downie, J. S., Laurier, C., Bay, M., and Ehmann, A. F. (2008a). The 2007 MIREX audio mood classification task: Lessons learned. In *Proceedings of the 9th International Society for Music Information Retrieval Conference (ISMIR)*, pages 462–467, Philadelphia, PA, USA. ISMIR.

Hu, Y., Koren, Y., and Volinsky, C. (2008b). Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining (ICDM)*, pages 263–272, Pisa, Italy. IEEE.

Hung, H.-T., Chen, Y.-H., Mayerl, M., Vötter, M., Zangerle, E., and Yang, Y.-H. (2019). MediaEval 2019 emotion and theme recognition task: A VQ-VAE based approach. In Larson, M. A., Hicks, S. A., Constantin, M. G., Bischke, B., Porter, A., Zhao, P., Lux, M., Quiros, L. C., Calandre, J., and Jones, G., editors, *Working Notes Proceedings of the MediaEval 2019 Workshop*, Sophia Antipolis, France. CEUR.

IFPI (2021). Global music report 2021.

Julià, C. F. and Jordà, S. (2009). SongExplorer: a tabletop application for exploring large collections of songs. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, pages 675–680, Kobe, Japan. ISMIR.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations (ICLR)*, San Diego, CA, USA. arXiv.

Knees, P., Schedl, M., and Goto, M. (2020). Intelligent user interfaces for music discovery. *Transactions of the International Society for Music Information Retrieval*, 3(1):165–179.

Knees, P., Schedl, M., Pohle, T., and Widmer, G. (2006). An innovative three-dimensional user interface for exploring music collections enriched. In *Proceedings of the 14th ACM International Conference on Multimedia (MM)*, pages 17–24, Santa Barbara, CA, USA. ACM.

Knox, D., Greer, T., Ma, B., Kuo, E., Somandepalli, K., and Narayanan, S. (2020). MediaEval 2020 emotion and theme recognition in music task: Loss function approaches for multi-label music tagging. In *Working Notes Proceedings of the MediaEval 2020 Workshop*, Online. CEUR.

Kohonen, T. (2001). *Self-organizing maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Heidelberg, 3 edition.

Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix Factorization Techniques for Recommender Systems. *Computer*, 42(8):30–37.

Koutini, K., Chowdhury, S., Haunschmid, V., Eghbal-Zadeh, H., and Widmer, G. (2019). Emotion and theme recognition in music with frequency-aware RF-regularized CNNs. In *Working Notes Proceedings of the MediaEval 2019 Workshop*, Sophia Antipolis, France. CEUR.

Law, E., West, K., Mandel, M. I., Bay, M., and Downie, J. S. (2009). Evaluation of algorithms using games: the case of music tagging. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, pages 387–392, Kobe, Japan. ISMIR.

Lee, J., Park, J., Kim, K. L., and Nam, J. (2017). Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms. In *Proceedings of the 14th Sound and Music Computing Conference (SMC)*, Espoo, Finland. Zenodo.

Leitich, S. and Topf, M. (2007). Globe of music - music library visualization using geosom. In *Proceedings of the 8th International Society for Music Information Retrieval Conference (ISMIR)*, pages 167–170, Vienna, Austria. ISMIR.

Li, A., Thom, J., Chandar, P., Hosey, C., Thomas, B. S., and Garcia-Gathright, J. (2019). Search mindsets: Understanding focused and non-focused information seeking in music search. In *The World Wide Web Conference (WWW)*, pages 2971–2977, San Francisco, CA, USA. ACM.

Liang, D., Zhan, M., and Ellis, D. P. W. (2015). Content-aware collaborative music recommendation using pre-trained neural networks. In *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, pages 295–301, Málaga, Spain. ISMIR.

Loeb, S. (1992). Architecting personalized delivery of multimedia information. *Communications of the ACM*, 35(12):39–47.

Maaten, L. v. d. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605.

Maltz, D. and Ehrlich, K. (1995). Pointing the way: Active collaborative filtering. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems (CHI)*, pages 202–209, Denver, CO, USA. ACM.

Martins, G. B., Papa, J. P., and Adeli, H. (2020). Deep learning techniques for recommender systems based on collaborative filtering. *Expert Systems*, 37(6).

Mayerl, M., Vötter, M., Hung, H.-T., Chen, B.-Y., Yang, Y.-H., and Zangerle, E. (2019). Recognizing song mood and theme using convolutional recurrent neural networks. In *Working Notes Proceedings of the MediaEval 2019 Workshop*, Sophia Antipolis, France. CEUR.

Mayerl, M., Vötter, M., Peintner, A., Specht, G., and Zangerle, E. (2021). Recognizing song mood and theme: Clustering-based ensembles. In *Working Notes Proceedings of the MediaEval 2021 Workshop*, Online. CEUR.

McInnes, L., Healy, J., and Melville, J. (2018). UMAP: uniform manifold approximation and projection for dimension reduction.

Mörchen, F., Ultsch, A., Nöcker, M., and Stamm, C. (2005). Databionic visualization of music collections according to perceptual distance. In *Proceedings of the 6th International Society for Music Information Retrieval Conference (ISMIR)*, pages 396–403, London, UK. ISMIR.

Nam, J., Choi, K., Lee, J., Chou, S.-Y., and Yang, Y.-H. (2019). Deep learning for audio-based music classification and tagging: teaching computers to distinguish rock from Bach. *IEEE Signal Processing Magazine*, 36(1):41–51.

Neumayer, R., Dittenbach, M., and Rauber, A. (2005). PlaySOM and PocketSOMPlayer, alternative interfaces to large music collections. In *Proceedings of the 6th International Society for Music Information Retrieval Conference (ISMIR)*, pages 618–623, London, UK. ISMIR.

Oramas, S., Nieto, O., Sordo, M., and Serra, X. (2017a). A deep multimodal approach for cold-start music recommendation. In *Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems (DLRS)*, pages 32–37, Como, Italy. ACM.

Oramas, S., Ostuni, V. C., Noia, T. D., Serra, X., and Sciascio, E. D. (2017b). Sound and music recommendation with knowledge graphs. *ACM Transactions on Intelligent Systems and Technology*, 8(2):1–21.

Pampalk, E., Dixon, S., and Widmer, G. (2004). Exploring music collections by browsing different views. *Computer Music Journal*, 28(2):49–62.

Pampalk, E., Rauber, A., and Merkl, D. (2002). Content-based organization and visualization of music archives. In *Proceedings of the 10th ACM International Conference on Multimedia (MM)*, pages 570–579, Juan-les-Pins, France. ACM.

Pariser, E. (2011). *The filter bubble: What the internet is hiding from you*. Penguin.

Pearson, K. (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

Pegoraro Santana, I. A., Pinhelli, F., Donini, J., Catharin, L., Mangolin, R. B., da Costa, Y. M. e. G., Delisandra Feltrim, V., and Domingues, M. A. (2020). Music4All: A new music database and its applications. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 399–404, Niterói, Brazil. IEEE.

Pham, P.-T., Huynh, M.-H., Nguyen, H.-D., and Tran, M.-T. (2021). SELAB-HCMUS at MediaEval 2021: Music theme and emotion classification with co-teaching training strategy. In *Working Notes Proceedings of the MediaEval 2021 Workshop*, Online. CEUR.

Pons, J., Nieto, O., Prockup, M., Schmidt, E. M., Ehmann, A. F., and Serra, X. (2018). End-to-end learning for music audio tagging at scale. In *Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR)*, pages 637–644, Paris, France. ISMIR.

Pons, J. and Serra, X. (2019). MusiCNN: Pre-trained convolutional neural networks for music audio tagging.

Porter, A., Bogdanov, D., Kaye, R., Tsukanov, R., and Serra, X. (2015). AcousticBrainz: A community platform for gathering music information obtained from audio. In *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, pages 786–792, Málaga, Spain. ISMIR.

Rajamani, S. T., Rajamani, K. T., and Schuller, B. W. (2020). Emotion and theme recognition in music using attention-based methods. In *Working Notes Proceedings of the MediaEval 2020 Workshop*, Online. CEUR.

Ricci, F., Rokach, L., and Shapira, B., editors (2022). *Recommender Systems Handbook*. Springer, 3 edition. In press.

Russell, J. A. (1980). A circumplex model of affect. *Journal of Personality and Social Psychology*, 39(6):1161–1178.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520, Salt Lake City, UT, USA. IEEE.

Schedl, M. and Flexer, A. (2012). Putting the user in the center of music information retrieval. In *Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR)*, pages 385–390, Porto, Portugal. ISMIR.

Schedl, M. and Hauger, D. (2015). Tailoring music recommendations to users by considering diversity, mainstreaminess, and novelty. In *Proceedings of the 38th ACM International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 947–950, Santiago, Chile. ACM.

Shardanand, U. (1994). Social information filtering for music recommendation. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, USA.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations (ICLR)*, San Diego, CA, USA. arXiv.

Smilkov, D., Thorat, N., Nicholson, C., Reif, E., Viégas, F. B., and Wattenberg, M. (2016). Embedding projector: Interactive visualization and interpretation of embeddings.

Soleymani, M., Caro, M. N., Schmidt, E. M., and Yang, Y.-H. (2013). The MediaEval 2013 brave new task: Emotion in music. In *Working Notes Proceedings of the MediaEval 2013 Workshop*, Barcelona, Spain. CEUR.

Sordo, M. (2012). *Semantic annotation of music collections: A computational approach*. PhD Thesis, Universitat Pompeu Fabra, Barcelona, Spain.

Stober, S. and Nürnberger, A. (2010). MusicGalaxy - an adaptive user-interface for exploratory music retrieval. In *Proceedings of the 7th Sound and Music Computing Conference (SMC)*, pages 23–30, Barcelona, Spain. Zenodo.

Sturm, B. L. (2014). The state of the art ten years after a state of the art: Future research in music information retrieval. *Journal of New Music Research*, 43(2):147–172.

Sukhavasi, M. and Adapa, S. (2019). Music theme recognition using CNN and self-attention. In *Working Notes Proceedings of the MediaEval 2019 Workshop*, Sophia Antipolis, France. CEUR.

Takahashi, T., Fukayama, S., and Goto, M. (2018). Instrudive: A music visualization system based on automatically recognized instrumentation. In *Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR)*, pages 561–568, Paris, France. ISMIR.

Tan, H. H. (2021). Semi-supervised music emotion recognition using noisy student training and harmonic pitch class profiles. In *Working Notes Proceedings of the MediaEval 2021 Workshop*, Online. CEUR.

Torrens, M., Hertzog, P., and Arcos, J. L. (2004). Visualizing and exploring personal music libraries. In *Proceedings of the 5th International Society for Music Information Retrieval Conference (ISMIR)*, Barcelona, Spain. ISMIR.

Tovstogan, P., Serra, X., and Bogdanov, D. (2020). Web interface for exploration of latent and tag spaces in music auto-tagging. In *Machine Learning for Music Discovery Workshop (ML4MD), International Conference on Machine Learning (ICML)*, Vienna, Austria.

Turian, J., Shier, J., Khan, H. R., Raj, B., Schuller, B. W., Steinmetz, C. J., Malloy, C., Tzanetakis, G., Velarde, G., McNally, K., Henry, M., Pinto, N., Noufi, C., Clough, C., Herremans, D., Fonseca, E., Engel, J., Salamon, J., Esling, P., Manocha, P., Watanabe, S., Jin, Z., and Bisk, Y. (2022). HEAR 2021: Holistic evaluation of audio representations.

Tzanetakis, G. (2001). Automatic musical genre classification of audio signals. In *Proceedings of the 2nd International Symposium on Music Information Retrieval (ISMIR)*, Bloomington, IN, USA. ISMIR.

Tzanetakis, G. and Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302.

Ultsch, A. (1992). Self-organizing neural networks for visualisation and classification. In *Information and Classification*, pages 307–313, Dortmund, Germany. Springer.

Vad, B., Boland, D., Williamson, J., Murray-Smith, R., and Steffensen, P. B. (2015). Design and evaluation of a probabilistic music projection interface. In *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, pages 134–140, Málaga, Spain. ISMIR.

van den Oord, A., Dieleman, S., and Schrauwen, B. (2013). Deep content-based music recommendation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 26, pages 2643–2651, Lake Tahoe, NV, USA. Curran Associates, Inc.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need.

In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pages 5998–6008, Long Beach, CA, USA. Curran Associates, Inc.

Vembu, S. and Baumann, S. (2004). A self-organizing map based knowledge discovery for music recommendation systems. In *2nd International Symposium on Computer Music Modeling and Retrieval (CMMR)*, volume 3310, pages 119–129, Esbjerg, Denmark. Springer.

Vötter, M., Mayerl, M., Specht, G., and Zangerle, E. (2020). Recognizing song mood and theme: Leveraging ensembles of tag groups. In *Working Notes Proceedings of the MediaEval 2020 Workshop*, Online. CEUR.

Wang, X. and Wang, Y. (2014). Improving content-based and hybrid music recommendation using deep learning. In *Proceedings of the 22nd ACM International Conference on Multimedia (MM)*, pages 627–636, Orlando, FL, USA. ACM.

Webber, W., Moffat, A., and Zobel, J. (2010). A similarity measure for indefinite rankings. *ACM Transactions on Information Systems*, 28(4):1–38.

Won, M., Choi, K., and Serra, X. (2021). Semi-supervised music tagging transformer. In *Proceedings of the 22nd International Society for Music Information Retrieval Conference (ISMIR)*, pages 769–776, Online. ISMIR.

Won, M., Chun, S., Nieto, O., and Serrc, X. (2020a). Data-driven harmonic filters for audio representation learning. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 536–540, Barcelona, Spain. IEEE.

Won, M., Ferraro, A., Bogdanov, D., and Serra, X. (2020b). Evaluation of CNN-based automatic music tagging models. In *Proceedings of the 17th Sound and Music Computing Conference (SMC)*, pages 331–337, Torino, Italy. Zenodo.

Wu, Y. and Takatsuka, M. (2006). Spherical self-organizing map using efficient indexed geodesic data structure. *Neural Networks*, 19(6-7):900–910.

Xie, Q., Luong, M.-T., Hovy, E., and Le, Q. V. (2020). Self-training with noisy student improves imagenet classification. In *2020 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, Seattle, WA, USA. IEEE.

Yang, Y.-H. and Chen, H. H. (2011). *Music Emotion Recognition*. CRC Press, 1 edition.

Yi, S., Wang, X., and Yamasaki, T. (2019). Emotion and theme recognition of music using convolutional neural networks. In *Working Notes Proceedings of the MediaEval 2019 Workshop*, Sophia Antipolis, France. CEUR.

Yoshii, K., Goto, M., Komatani, K., Ogata, T., and Okuno, H. G. (2006). Hybrid collaborative and content-based music recommendation using probabilistic model with latent user preferences. In *Proceedings of the 7th International Society for Music Information Retrieval Conference (ISMIR)*, pages 296–301, Victoria, Canada. ISMIR.

Zentner, M., Grandjean, D., and Scherer, K. R. (2008). Emotions evoked by the sound of music: Characterization, classification, and measurement. *Emotion*, 8(4):494–521.

Zhao, Y. and Guo, J. (2021). MusiCoder: A universal music-acoustic encoder based on transformer. In *27th International Conference on Multimedia Modeling (MMM)*, volume 12572, pages 417–429, Prague, Czech Republic. Springer.

# Appendix A

# PUBLICATIONS BY AUTHOR

## Conference papers

Tovstogan P., Serra X., & Bogdanov D. (2022). Visualization of Deep Audio Embeddings for Music Exploration and Rediscovery. *Proceedings of the 19th Sound and Music Computing Conference (SMC)*

Tovstogan P., Serra X., & Bogdanov D. (2022). Similarity of Nearest-Neighbor Query Results in Deep Latent Spaces. *Proceedings of the 19th Sound and Music Computing Conference (SMC)*

## Workshop papers

Tovstogan, P., Bogdanov, D., & Porter, A. (2021). MediaEval 2021: Emotion and theme recognition in music using Jamendo. *Working Notes Proceedings of the MediaEval 2021 Workshop*

Bogdanov, D., Porter, A., Tovstogan, P., & Won, M. (2020). MediaEval 2020: Emotion and theme recognition in music using Jamendo. *Working Notes Proceedings of the MediaEval 2020 Workshop*

Tovstogan P., Serra X., & Bogdanov D. (2020). Web Interface for Exploration of Latent and Tag Spaces in Music Auto-Tagging. In *Machine Learning for Media Discovery Workshop (ML4MD), 37th International Conference on Machine Learning (ICML)*

Bogdanov, D., Porter, A., Tovstogan, P., & Won, M. (2019). MediaEval 2019: Emotion and theme recognition in music using Jamendo. *Working Notes Proceedings of the MediaEval 2019 Workshop*

Bogdanov, D., Won M., Tovstogan P., Porter A., & Serra X. (2019). The MTG-Jamendo Dataset for Automatic Music Tagging. *In Machine Learning for Music Discovery Workshop (ML4MD), 36th International Conference on Machine Learning (ICML)*

# Appendix B

# SURVEY ON MUSIC LISTENING, DISCOVERY AND EXPLORATION BEHAVIOR

This is a survey that was shared on the internet.

Background questions:

- What gender do you identify as? Man / Woman / Prefer not to say / Other
- What is your age? 18–24 / 25–34 / 35–44 / 45–54 / 55–65 / Prefer not to say
- Which country do you currently live in? Prefer not to say / 204 countries

General music questions:

- Please select how often do you engage in the following activities: (Never / Rarely / Sometimes / Very often / Always)
  - I write about music on social media
  - I keep track of new music that I come across (e.g. new artists or recordings)
  - I read or search the internet for things related to music

  - I do music-related activities in my free time
  - I try to find out more about music I'm not familiar with
  - I pick certain music to motivate or excite me
  - I listen to music to trigger the associated memories / put myself into associated mood
- How would you describe your musical background?
  - I don't have any musical training
  - I have some musical training
  - I am a hobbyist/amateur musician
  - I am a professional musician
- On average, how many hours per day do you spend listening to music? Less than 1 / 1–2 / 3–4 / 5–6 / 7–8 / More than 8
- Out of those hours, how many hours do you spend listening actively? (You put the music not just in the background, and you are not doing anything else at the same time that takes away your focus. For example, listening during the commute, before going to sleep, checking out new album.) Less than 0.5 / 0.5–1 / 1–2 / 3–4 / 5–6 / 7–8 / More than 8
- How do you get access to the music you listen to? (If you listen to music from multiple places, indicate where do you spend the most time)
  - Streaming (e.g. Spotify, Apple Music, Deezer, YouTube, etc.)
  - Owned music (e.g. digital, vinyls, BandCamp)
  - Other
- If you use streaming services, what sources do you usually use to listen to music? (If you don't use streaming services, you can select the options that you imagine yourself using.)
  - I don't use streaming services
  - Something quick from the home page
  - My library/artists that I follow and know
  - Playlists created by me
  - Playlists created by other users
  - Playlists curated by platform (e.g. Evening Chillout, Rock Classics, Essential Trap)

  - Algorithmically generated playlists of your music (e.g. your daily/genre/artist mixes)
  - Algorithmically generated playlists of new music (e.g. discover weekly, new releases)
  - Algorithmically recommended albums/artists (e.g. because you liked X, based on your activity)
  - Other
- Do you usually listen to playlists made by others? What kind? (This question refers to situation when you know what you want to listen to.)
  - I don't typically listen to playlists
  - Based on genre / style (e.g. rock, pop, metal, reggae, deep house, symphonic metal)
  - Based on culture / country of origin, location / regional scene (e.g. oriental metal, J-Pop, bands from Barcelona)
  - Based on moods / themes (e.g. chill, party, sleep, workout, melancholic)
  - Based on instrumentation (e.g. female vocal, electric guitar, sax, electronic synths)
  - Based on decade (e.g. 80s, 90s, 00s, 10s)
  - Based on editorial metadata (e.g. artists, music producers, recording labels)
  - Other
- How large is your personal music library? (if you have any) (You can provide answer in whatever units that you are comfortable, e.g. 50 artists, 2000 tracks, 500 hours. Answer this question in terms of whatever you consider your "personal music library", e.g. followed/saved artists/albums/tracks on streaming services, bought records on Bandcamp, music stored on hard drive, number of physical records.)

Music exploration and discovery:

- How often do you have a desire to listen to new music?
- How often would you like to listen to something from your collection/library that you haven't listened in a long time?

- How often do you ACTUALLY listen to something from your collection/library that you haven't listened in a long time?
  - Every day
  - Once or several times per week
  - Once or several times per month
  - Once or several times per year
  - Never or almost never
- Why do you think you don't listen to those parts of your collection/library as often as you would like? (if your answers to previous two questions are different)
- What are your go-to sources to discover new music?
  - Streaming platform discover functionality
  - Social recommendations (e.g. asking a friend)
  - Music journalism (e.g. review articles, best albums of the year)
  - Influencers or highly reputed journalists that I follow
  - Music identification apps (e.g. Shazam)
  - Music stores (e.g., new arrivals, charts, selling right now on Bandcamp, album of the day)
  - Other
- What is your usual music discovery and exploration strategy? (For example, you might want to find some music for a specific context or genre that you heard about. Everyone is different, feel free to describe your strategy.)
  - I know what I am looking for and/or research a specific topic (e.g. trance metal, side project of artist X)
  - I have a vague idea of what I want to listen (e.g. something jazzy, melacholic)
  - I have no idea, usually I am pretty open
  - Other
- When you are looking for new music, which of these types of information is the most useful for you to explore? (Probably you use mix of these terms, please select the option that is the most important to you)

  - Genre / style (e.g. rock, pop, metal, reggae, deep house, symphonic metal)
  - Culture / country of origin, location / regional scene (e.g. oriental metal, J-Pop, bands from Barcelona)
  - Moods / themes (e.g. chill, party, sleep, workout, melancholic)
  - Instruments (e.g. female vocal, electric guitar, sax, electronic synths)
  - Decade (e.g. 80s, 90s, 00s, 10s)
  - Editorial metadata (e.g. relations between artists, music producers, recording labels)
  - Other
- How many new artists have you discovered in last year? 0 / 1–2 / 3–4 / 5–10 / 10–20 / More than 20
- In which context do you usually like to explore and discover new music? (E.g. jogging, workout, in commute, chilling at home)
- What is your motivation for music exploration and discovery?
  - Curiosity
  - Getting tired of listening to the same music
  - Getting more of my favorite type of music
  - Desire to learn about new music
  - Keeping up with trends
  - Pushing myself out of comfort zone
  - Connecting with people
  - Other
- Talking about modern music streaming services, do you agree or disagree with the following statements: (Strongly disagree / Disagree / Neither agree nor disagree / Agree / Strongly agree)
  - There are many options for music exploration and discovery
  - I am satisfied with the options for music exploration and discovery that are available
  - The way terms ”music exploration and discovery” are used aligns well with my perception
  - I would like more functionality to rediscover my library

  - I would like to interact with my library in more ways than current systems allow
  - I can usually quickly find the music that I want to listen to
  - I have discovered music that is different from what I usually listen to through recommendations
  - There should be more recommendations outside of my comfort zone
  - It is easy for me to get an overview and manage my library
- Do you have any additional comments or thoughts to share regarding music exploration and discovery?

# Appendix C

# INTERVIEW QUESTIONNAIRE

This is a questionnaire that was presented to the users after the semi-structured interview.

Background questions:

- Name
- Age
- Gender: Man / Woman / Prefer not to say / Other
- Do you have any form of musical training (either formal or just classes): Yes / No
- If you answered yes to the previous question, how many years?
- How many hours on average do you listen to music per day (both actively and in the background)?
- How many hours on average do you ACTIVELY listen to music per day?
- Where do you usually listen to music: Streaming platforms / Personal collection; Other
- Which percentage of the time that you listen to music do you listen to playlists (including created by others, or algorithmically generated)?
- How often do you create playlists?

- How often do you feel the desire to listen to something from your collection that you haven't listened in a while?
  - Every day
  - Once or several times per week
  - Once or several times per month
  - Once or several times per year
  - Never or almost never

Questions about the system:

- Which features of the interface is your favorite? (select at most 3)
- Which features of the interface you dislike or have troubles with? (if any)
  - Selecting tags to visualize
  - Selecting artists to visualize
  - Seeing different architecture/dataset/layer side by side
  - Seeing different projections side by side of the same model
  - Reducing the number of segments visualized to see more tracks at the same time
  - Highlighting particular artists/albums/tags/tracks
  - Clicking/hovering the points to listen to the segment
  - Coupled labels on graphs
  - Coupled selection of graphs
- Did you find any particular combination of architecture/dataset/layer/projection more interesting or meaningful than others: Yes / No
- Which combination of architecture/dataset/layer/projection do you like the most?
- Select whether you agree or disagree with the following statements: (Strongly disagree / Disagree / Neither agree nor disagree / Agree / Strongly agree)
  - I like interacting with the system
  - I have a preference for a particular architecture/dataset/layer/projection
  - I found the system more useful or interesting to explore my music collection comparing to regular browsing by metadata (artists/albums/tracks/genres)
  - I found the system more useful or interesting to explore my

music collection compared to random shuffle
- I think that the visualizations captured a good overview of my library
- I think that the visualizations managed to capture the similarity between the track segments
- I feel that I discovered some interesting connections between the tracks in my library that were not obvious to me before
- This system made me want to listen to some parts of my music collection that I haven't listened to in a while
- I would like to sometimes use this system for playlist creation
- I would like to use this system to get ideas when I am not sure what to listen to next
- I feel that interacting with the system is a rewarding experience for me
- I think that interacting with the system is an engaging experience for me

If you have any comments or suggestions, please write them here.