

Improving Sound Retrieval in Large Collaborative Collections

Xavier Favory

TESI DOCTORAL UPF / year 2021

THESIS SUPERVISORS

Dr. Xavier Serra i Casals
Dr. Frederic Font Corbera

Dept. of Information and Communication Technologies
Universitat Pompeu Fabra, Barcelona



Acknowledgments

I can't express enough my gratitude towards my supervisors for their support. To Xavier Serra and Frederic Font, it was an honor to have had the opportunity to work under your supervision. Thank you for trusting me all along the course of my PhD. Infinite thanks to Konstantinos Drossos, who was able to push me and support me in my last year of effort.

Special thanks to Jordi Pons, who never stopped believing in me, advised me and showed me how simple research was. To Eduardo Fonseca, who knew how to listen to my ideas and challenge me to accomplish them. Sincere thanks to Nicolas Obin, Jeremie Garcia and Jean Bresson from IRCAM, with whom I've started research and who taught me a lot. I also would like to acknowledge the support and companionship of all the people I've met in the UPF, including Alastair, Sergio Oramas, Dmitry, Andrés, Albin, Rafa, Olga, Rong, Marius, Georgi, Pritish, António, Pablo Alonso, Pablo Zinemanas, Minz, Furkan, Juan, Lorenzo, Helena, Emir, Rachit, Philip, Sonia, Cristina, Ajay, Sankalp, Swapnil, Sertan, Vsevolod, Oriol, Fabio, Angel Blanco, Angel Faraldo, Panos, Juanjo, Alvaro, David, Sergio Giraldo, Merlijn, Sergi, Maria, Federico and Rasoul. Additionally, thank you to all the people I've met in Barcelona, particularly to Pau, Julia, Pep, Neus, Mariona, Ignasi, Cape, Javi and Willy.

Finally, I would like to thank all my family, in particular my parents, for supporting me in this PhD and during my entire life. An absolute thank you to Ceci, for her support, love and help.

Abstract

Capturing sounds on a recording medium to enable their preservation and reproduction started to be possible during the industrial revolution of the 19th century, originally achieved through mechanic and acoustic devices, and later electronic and magnetic ones. Eventually, the digital age of the mid-20th century brought about the democratization of recording and reproduction devices, as well as accessible ways of storing and sharing content. As a consequence, massive collections of audio samples are nowadays increasingly available online, some of which are created collaboratively thanks to sharing platforms. This content has become essential for entertainment media, such as movies, music, video games, and for human-machine interaction. Nonetheless, given the amount and diversity of the content, exploring, searching and retrieving from collaborative collections becomes increasingly challenging. Methods for automatically organizing content, and facilitating its retrieval therefore become more and more necessary, creating an opportunity for novel Information Retrieval approaches.

This thesis aims at improving the retrieval of sounds in large collaborative collections, and does so from different perspectives. We first investigate data collection methodologies for creating large and sustainable audio datasets, including the design and development of a website and an annotation tool to engage users in the collaborative process of dataset creation. Additionally, we focus on improving the manual annotation of audio samples when using large taxonomies. This calls for specialized tools to assist users towards providing exhaustive and consistent annotations. This produced a number of publicly available large-scale datasets for developing and evaluating machine listening models. From another perspective, we propose novel methods for learning audio representations, suitable for diverse machine learning applications, by taking advantage of large amounts of online content and its metadata. We then investigate the problem of unsupervised classification by first identifying which type of audio features are suited for clustering the wide variety of sounds present in online collections. Finally, we focus on Search Results Clustering, an

approach that organizes the search results into coherent groups. This research improved the retrieval of sounds from large collections, namely through facilitating exploration and interaction with search results.

Resumen

A mediados del siglo XIX, y más precisamente durante la segunda Revolución Industrial, comenzó a ser posible la captura de sonidos gracias a un soporte de grabación, permitiendo su conservación y su reproducción. En un principio, esto se logró gracias a dispositivos mecánicos y acústicos, y posteriormente éstos fueron electrónicos y magnéticos. Finalmente, a mediados del siglo XX, la era digital trajo consigo la democratización de los dispositivos de grabación y de reproducción, así como el acceso a otras formas de almacenamiento y de compartimiento de contenido. Como consecuencia, hoy en día, aumentan las colecciones disponibles en línea. Se trata de colecciones masivas de muestras de audio, algunas de las cuales se crean de forma colaborativa, gracias a las plataformas de intercambio. Este contenido ha llegado a ser imprescindible para los medios de entretenimiento, como películas, música, videojuegos y para la interacción hombre-máquina. No obstante, dada la cantidad y la diversidad existentes, explorar, buscar y recuperar contenido de colecciones colaborativas es cada vez más difícil. Así, los métodos para organizar automáticamente el contenido y facilitar su recuperación, son cada vez más necesarios. Esta situación es una oportunidad para el estudio de enfoques novedosos cuyo objetivo es la recuperación de información.

Desde diferentes perspectivas, esta tesis tiene como objetivo el facilitar la recuperación de sonidos ubicados en grandes colecciones colaborativas. En primer lugar, investigamos los métodos de recopilación de datos para crear grandes conjuntos sostenibles de datos de audio, incluido el diseño y el desarrollo de una aplicación web y de una herramienta de anotación para involucrar a los usuarios en el proceso colaborativo de creación de conjuntos de datos. Además, nuestro trabajo se enfoca hacia la mejora de la anotación manual de muestras de audio, cuando usamos taxonomías grandes. Esta operación requiere herramientas especializadas que faciliten las anotaciones exhaustivas y consistentes. El resultado es la producción de una serie de conjuntos de datos a gran escala disponibles, a nivel público, que permiten desarrollar y evaluar modelos de apren-

dizaje de máquinas. Desde una perspectiva original, proponemos métodos novedosos y adecuados para, en primer lugar, aprender representaciones de audio y, en segundo lugar, para realizar diversas aplicaciones de aprendizaje automático, aprovechando grandes cantidades de contenido en línea y sus metadatos. En segundo lugar, investigamos el problema de la clasificación sin supervisión, identificando qué tipo de características de audio son las adecuadas para agrupar la amplia variedad de sonidos presentes en las colecciones en línea. Por último, nos centramos en la agrupación de resultados de búsqueda, un enfoque que organiza los resultados en grupos coherentes. Esta investigación facilita la recuperación de sonidos de grandes colecciones, principalmente, al facilitar la exploración y la interacción con los resultados de búsqueda.

Contents

List of figures	xiv
List of tables	xvii
1 Introduction	1
1.1 Motivations	2
1.2 Sound retrieval	3
1.2.1 Text-based and content-based search	3
1.2.2 The role of user interfaces	5
1.3 Innovative methods	6
1.3.1 Classification	7
1.3.2 Audio representation	8
1.3.3 Clustering	9
1.4 Objectives and outline of the thesis	10
2 Data Collection	15
2.1 Introduction	16
2.1.1 Annotation and taxonomies	16
2.1.2 Datasets	17
2.1.3 Dataset desirata	18
2.2 The Freesound Annotator platform	20
2.2.1 FSD: a dataset for general machine listening	21
2.2.2 Functionalities of the platform	22
2.3 Outcomes and conclusion	29

3	Improving the Manual Annotation of Audio Content	31
3.1	Introduction	32
3.2	Motivations	35
3.2.1	Audioset	35
3.2.2	Motivating new annotation tools	36
3.3	The annotation tools	37
3.3.1	Generate annotations	38
3.3.2	Refine annotations	38
3.4	Evaluation	41
3.4.1	Methodology	43
3.4.2	Results and discussion	43
3.5	Conclusion	46
4	Audio Feature Performance Comparison for Unsupervised Sound Classification	49
4.1	Introduction	50
4.2	Related work	51
4.2.1	Audio features	51
4.2.2	Clustering	53
4.2.3	Clustering validation	55
4.3	Experiment	56
4.3.1	Clustering methods	56
4.3.2	Audio features	57
4.3.3	Datasets	58
4.4	Results	58
4.4.1	Automatic evaluation	58
4.4.2	Qualitative evaluation	64
4.4.3	Discussion	69
4.5	Conclusion	78
5	Co-Aligned Autoencoders for Learning Semantically Enriched Audio Representations	80
5.1	Introduction	81
5.2	Co-aligned autoencoders	83

5.2.1	Learning low-level audio and semantic features . . .	85
5.2.2	Alignment of acoustic and semantic features . . .	87
5.3	Experiment	88
5.3.1	Pre-training dataset and data pre-processing . . .	89
5.3.2	Utilized hyper-parameters, training procedure, and models	90
5.3.3	Downstream classification tasks	91
5.3.4	Models from the literature	92
5.3.5	Correlation analysis with acoustic features	93
5.4	Results	93
5.4.1	Classification performance	93
5.4.2	Correlation analysis	95
5.4.3	Clustering performance	96
5.5	Conclusions	97
6	Learning Contextual Tag Embeddings for Cross-modal Alignment of Audio and Tags	101
6.1	Introduction	102
6.2	Proposed method	103
6.2.1	Audio encoding and decoding	104
6.2.2	Multi-head, self-attention tags encoding	105
6.2.3	Cross-modal alignment and optimization	106
6.3	Evaluation	107
6.3.1	Pre-training dataset and data pre-processing . . .	107
6.3.2	Audio-based classification	108
6.4	Results	110
6.5	Conclusion	111
7	Search Results Clustering	113
7.1	Introduction	115
7.2	Related work	121
7.3	Proposed approach	124
7.3.1	Audio features	124
7.3.2	Graph-based clustering	126

7.3.3	Discarding low quality clusters	126
7.3.4	Selection of cluster representative examples	127
7.3.5	User interfaces	127
7.4	Feature performance comparison	130
7.4.1	Internal validation	130
7.4.2	External validation	131
7.4.3	Results	133
7.5	User evaluation	134
7.5.1	Methodology	135
7.5.2	Results and discussion	136
7.6	Conclusion	140
8	Conclusions	145
8.1	Summary of contributions	146
8.2	Building high-quality datasets	148
8.3	Audio representation learning	150
8.4	Feature performance for clustering	152
8.5	Search Results Clustering	153

List of Figures

1.1	Diagram representing the topics covered in this thesis. An arrow indicates that the outgoing element contributes to the incoming element.	12
2.1	Screenshot of the familiarisation interface of the Freesound Datasets platform validation task	22
2.2	Screenshot of the interface when performing the validation task.	23
2.3	Screenshot of the annotator ranking during an annotation event.	25
2.4	Screenshot of a category exploration table, with the error report tracking.	26
2.5	Screenshot a section of the mapping rule monitoring page which allow maintainers to retrieve new candidate annotations.	27
2.6	Screenshot of inspection section of the mapping rule monitoring page for a specific category and submitted mapping.	28
3.1	Representation of a small part of the AudioSet Ontology hierarchy.	36
3.2	Screenshot of the Audio Commons Manual Annotator	39
3.3	Screenshot of the Audio Commons Manual Annotator taxonomy table, showing the descriptions and examples of “Sigh” and “Groan”, together with their hierarchy location	40

3.4	Screenshot of the Audio Commons Refinement Annotator displaying a sound sample and its three suggested label paths	41
3.5	Screenshot of the Audio Commons Refinement Annotator showing a dropdown displaying the children categories of “Guitar”	42
3.6	Screenshot of the Audio Commons Refinement Annotator showing the description and examples of the “Guitar” category in a popup	42
4.1	Box-plots of the Adjuster Mutual Information scores re-grouped by dataset family and for all the datasets.	64
4.2	Visualisation of the clustered graph for the "Brass instrument" dataset using the AudioSet features. The graphs for different datasets and features can be explored from a browser at this url: https://xavierfav.github.io/feature-comparison-clustering/web-visu/	65
4.3	First two components of the PCA decomposition of the audioset embeddings for the different datasets. The first two plots’ colors represent clusters obtained with the KNN and the K-means approaches respectively, the last one displays the ground truth labels.	74
5.1	Illustration of our proposed method. \mathbf{Z}_a and \mathbf{z}_t are aligned through maximizing their agreement and, at the same time, are used for reconstructing back the original inputs.	84
6.1	Illustration of our method. ϕ_a and ϕ_w are aligned by maximizing their agreement through contrastive learning and, at the same time, \mathbf{Z}_a is used for reconstructing back the original spectrogram input. Word embeddings are passed through a multi-head scaled dot-product self-attention layer in order to build higher-level semantic vectors that are finally aggregated into a single vector ϕ_w	105

7.1	Screenshot of the publicly available web search results clustering interface using Carrot2 requested with the “pandas” query.	118
7.2	Screenshot of the the Yippy search engine requested with the “pandas” query.	119
7.3	Screenshot of the the Google image search engine requested with the “pandas” query.	120
7.4	Diagram representing the steps of our clustering engine.	124
7.5	Page displaying the result of the query <i>glass</i> of the <i>cluster #1</i> . Clicking on a cluster facet on the right applies a cluster filter. Three labels are shown for each cluster, together with the number of sounds they contain.	128
7.6	The graphical 2D visualisation of sounds retrieved with the query <i>guitar</i> . Each circle represents a sound. Placing the mouse on one will play the associated sound. Clicking on it displays some information at the top of the screen and highlights neighbor nodes.	129
7.7	Diagram representing the steps of our internal evaluation making use of user-provided tags. The Calinski-Harabasz Index is calculated between the labels corresponding to the obtained clusters and the features derived from the sound tags. This evaluation is performed on the results of the 1000 most popular queries performed by Freesound users.	132
7.8	Results from the <i>guitar</i> query in the search engine. Clustering facets behave like the traditional one, enabling users to combine different types of facets to further narrow down the search results.	142
7.9	The 2D visualisation displaying the kNN-graph for the <i>guitar</i> query.	143

List of Tables

4.1	Dataset families content.	59
4.2	Average performance (AMI) across the different dataset families of the K-means and the KNN-Graph clustering methods with the different features. An AMI close to 0 corresponds to a random partition while perfect matches gives 1 AMI.	60
5.1	Average mean accuracies for SER, MGC, and MIC. Additional performances are taken from the literature (Cramer et al., 2019; Salamon & Bello, 2017; Pons & Serra, 2019b; Lee et al., 2018; Ramires & Serra, 2019).	94
5.2	CCA correlation scores between the embeddings model outputs and some acoustic features statistics.	96
5.3	Average performance (AMI) across the different dataset families of the K-means and the KNN-Graph clustering methods with the different features. An AMI close to 0 corresponds to a random partition while perfect matches gives 1 AMI.	98
6.1	Average mean accuracy for SER, MGC, and MIC. Additionally performances on US8K dataset using a tag-based classifier are reported in the last column.	111
7.1	The different features compared in this work.	125

7.2	Clustering validity score (Calinski-Harabasz Index) using the different feature sets. Mean and standard deviation is calculated on the performance of the clustering of the results from the top 1000 most popular queries in Freesound. The pruning column corresponds to the validity score when discarding the cluster with the lowest confidence score defined in Section 3.3.	134
7.3	Average performance (AMI) across the different dataset with the different features. An AMI close to 0 corresponds to a random partition while perfect matches gives 1 AMI. The pruning column corresponds to the performance when discarding the cluster with the lowest confidence score defined in Section 7.3.3.	135

Chapter 1

Introduction

1.1 Motivations

The rapid advance in music technology, including hardware and information technologies, has made digital audio a major part of modern everyday life. Sound collections are often used by sound designers for making movies, video-games and other media, an immersive multisensorial experience. Therefore, the need to interact effectively with increasingly large digital audio collections is growing in many applications. Whenever we hear footsteps, a door closing or thunder in a movie clip, it almost always originates from large libraries of sound effects. In computer games, sound is a valuable and necessary component for producing emotional reactions. Moreover, composers and arrangers of music frequently use collections of sounds to create musical pieces. From simple clips to more complex music loops, electronic composers have numerous possibilities to arrange, transform and finally create new pieces of music. As an other example, audio is intensively used in human-computer interaction to provide richer and more robust and inclusive environments, compared to just having the graphic representation. Audio feedback can complement graphics and text for reinforcing a concept in the user's mind. Efficient approaches for retrieving and interacting with audio content from large collections will therefore empower a large and diverse group of creators and consumers in their usability and creative needs and experiences.

Nowadays, the creation and generation of audio content is widely facilitated, without the need of extensive resources and equipment. Thanks to online sharing platforms, this content can be easily and freely shared, resulting in an exponential growth in available data. In these platforms, sound collections are collectively generated by their users instead of coming from professional studios, which poses challenges for their organization and retrieval as the collections are typically not uniformly labeled and categorized. Moreover, the increasing size of collaborative collections is one of the main challenges in the management and retrieval of its sounds.

Effective retrieval from large collections requires the content to be organized and accompanied by the necessary information. The information that complements a sound file, also known as metadata, can be either gen-

erated automatically, or provided by the user. Most importantly, metadata includes annotations in the form of textual representation and descriptions. Online sharing sites often rely on the creator of the sound for annotating its content. Each platform has its own content description approach, typically consisting of a form with several predefined fields. The user then specifies e.g. the musical genre, or information about the sound source(s). Other strategies employ a more flexible description form, where the user can provide a textual description, together with a list of labels. These labels are known as *tags* and are widely used in various multimedia sharing platforms, such as Flickr, Vimeo, Soundcloud, Last.fm or Freesound. Since it is provided by many users with different backgrounds, crowd-generated content is non-uniformly annotated when compared to that of commercial libraries, which employ trained experts (Font et al., 2018). Efficient indexing and curation of user-generated content, whose metadata do not comply to a standard format, is therefore very challenging. Moreover, given the pace at which new audio content is nowadays generated, human expert annotation simply does not scale. These concrete and open challenges, together with the potential of recent advances in machine learning, motivate the research presented in this thesis.

1.2 Sound retrieval

As with any type of multimedia content, sound retrieval involves mainly indexing and searching. These techniques lie in the scope of *Information Retrieval*, defined as “a field concerned with the structure, analysis, organization, storage, searching, and retrieval of information”, according to one of its pioneers (Salton, 1968). In this section we’ll look into different aspects of the field of Information Retrieval when applied to sound.

1.2.1 Text-based and content-based search

Search techniques can be divided into keyword-based retrieval (Jacobs, 2014) and content-based retrieval (Lew et al., 2006). The former cor-

responds to the use of text-based search engines, which rely on the accompanying text elements of sounds such as title, description or tags. Content-based retrieval consists in performing search directly using the content itself, or more particularly, representations derived from it. In both cases, the retrieval engine can benefit from automatic methods that either augment the metadata with new annotations, or provide a numerical representation that reflects high-level characteristics of the sounds. The latter can be used with a distance metric for performing queries by examples, for instance.

The majority of current platforms rely heavily on text-based search engines to retrieve sounds. Generally, audio sample providers employ predefined categories to annotate the content, e.g. *dog bark*, *guitar* and *loop*. Organizing a collection through such a controlled set of categories significantly facilitates the indexing and retrieval of its sounds by allowing to display sounds to users organized in such controlled categories. This is particularly suitable when the collections are relatively small and focus on specific types of sounds, i.e. a smaller number of categories. For example, many professional libraries specialize into *animal*, *urban*, or even *fantasy creature* sounds. As a consequence, searching for sounds in such reduced-scope collections is easier than exploring more comprehensive ones, containing mixed types of sounds.

Alternatives to professional libraries, such as the ones available in collaborative platforms, are usually more diverse in the nature and quality of their content. Resources are often freely annotated, rather than following a set of predefined categories. It has the advantage of being less demanding for the contributing user, as it gives her the freedom to employ her own vocabulary. However, this hinders the performance of the search engine (Furnas et al., 1987). If the user-formulated query does not include the exact terms used in the description or tags of the sounds, according to which the content is indexed, the system will potentially not retrieve the relevant sounds. In other words, this will affect the *recall* of the system. In addition, if the metadata and/or query text are not precise enough or include erroneous terms, the system may retrieve irrelevant sounds, thus decreasing the *precision* of the search. In collaborative collections, these

types of vocabulary miss-matches can be frequent, and represent a significant bottleneck for traditional text-based search approaches.

There are several possible solutions to these problems. A popular category of approaches rely on comprehensive human knowledge in order to appropriately grasp the semantic content of the text to improve search. For example, one strategy is to expand the queries with terms or features that are synonyms or related to the user input. This technique is known as *query expansion*, and can improve retrieval performance by bridging the gap in the vocabulary used in the query and that used in the documents (Carpineto & Romano, 2012). In addition, *word sense disambiguation* is employed in order to tackle polysemy (Navigli, 2009). Natural language processing approaches to improve search are however outside of the scope of this thesis. Another way to tackle this challenge is to concentrate on content-based approaches, such as automatic annotation and classification of sounds (Martinez et al., 2009; Turnbull et al., 2008). When relying on the content, the search performance becomes less dependent on the quality of the user-provided annotations.

1.2.2 The role of user interfaces

Human Computer Interaction research also plays an important role in Information Retrieval. Graphical user interfaces enable the creation of powerful tools that combine human and machine capabilities. Existing works emphasized on automatic algorithms are rarely concerned with how users can interact with these algorithms. However, human computer interaction is a very important aspect in the creation of effective retrieval systems. Most of the tools used by sound designer and musicians offer limited capabilities for browsing large collections of audio files. Digital audio workstations typically display filenames and sometimes some forms of tag to explore personal collections of sounds. Professional collections are grouped by type and structured with handcrafted organization of concepts which may allow text-based retrieval method to be efficient enough. In the case of online large collections, tag clouds have been introduced and provide to users a valuable visual interface to browse web collec-

tions (Hassan-Montero & Herrero-Solana, 2006). They list the most popular tags present in the data collection and allow to construct queries by selecting tags. Even if tags are sometimes disorganized due to their personal and subjective origin, they directly reflect the vocabulary of users, enabling matching users' real needs and language. Another aspect, called faceted search, allows to browse the collection in a multi-step refinement process by selecting categories, proprieties, attributes or characteristics of the content (Fagan, 2010). Using facets to structure the information is commonly used for providing an intuitive interface for browsing collections using metadata.

Alternatives based on content-oriented methods for the browsing and the exploration of sound collections have been investigated. They often consist in projecting the numerical audio features into a small dimensional space where similar sounds are close from one another. This type of approaches can be considered appealing to some, by providing interaction methods that can make the process of browsing content more exciting and joyful (Schedl, 2017).

1.3 Innovative methods

Advancements in Information Retrieval and Machine Learning have a great potential for improving sound retrieval methods in different ways. First, by using automatic annotation tools, online content can be augmented with annotations that then can be indexed and used to improve the text-based retrieval engine. As a requirement for automatic methods, having good numerical audio representations are a key to their success. Finally, clustering is considered suitable for data exploration since it can automatically discover hidden patterns in large amount of data (Berkhin, 2006).

1.3.1 Classification

In the past decades, automatic content description methods have proliferated and can be used, with different accuracies, for detecting semantic concepts from low-level features derived from the content digital representation. However, there is a persistent *semantic gap* (Celma et al., 2006) produced by the lack of accordance between the information that can be extracted from the data and the interpretation that the same data has for a user. Successful automatic description methods are based on approaches that often rely on a lot of data for training and evaluation. As a consequence, manual generation of content description is of high importance for the realisation of intelligent systems able to produce meaningful automatic content descriptions, and to make steps towards reducing the *semantic gap*.

In the context of machine learning, classification was typically performed by using a set of hand-crafted features that would then be input to a classification model that would learn to assign labels to content. Recently, deep learning has been able to leverage massive amount of data in order to both learn audio representations and train classifiers. However, successful approaches often make use of large annotated corpus which requires to have previously selected a set of classes. When dealing with online audio data comprising a large range of type of sounds, these set of classes needs to be large enough in order to accurately describe the content. The largest public taxonomy of concepts used in audio is AudioSet and consists of more than 600 classes, organized in a hierarchy (Gemmeke et al., 2017). In the image field however, where deep learning has enabled huge improvements, the size of these taxonomies is much larger and can include thousands of concepts. However, generating ground truth with categories drawn from large sets of categories is difficult, since human annotators need to have a good knowledge about them in order to use them appropriately. To address this issue, better tools for manual annotation should be developed that better assist users in dealing with large taxonomies in the ground truth generation process. This thesis proposes to leverage online audio content to build larger audio datasets that can be

used to train general-purpose audio classifiers. To do so, novel interfaces that intelligently guides users in the process of annotating audio content are investigated.

1.3.2 Audio representation

Audio representations were originally created by relying on hand-crafted numerical features designed using domain knowledge about invariances in classes and as an attempt to extract audio characteristics that might be perceptually relevant. Some are derived directly from the time domain audio representation, while others are derived from spectral representations of the sounds which is mostly motivated by the fact that human perception widely relies on the frequency content of sound signals. These sets of features allow representing audio in a high-dimensional space, which enable the use of automatic methods for describing the content. These features have been successful to some extent in a number of applications such as musical instrument classification (Eronen & Klapuri, 2000), music genre recognition (Tzanetakis & Cook, 2002), acoustic scene classification (Barchiesi et al., 2015), sound event recognition (Janvier et al., 2012) or clustering (Herrera-Boyer et al., 2003). Moreover, they allow to perform content-based retrieval by for instance proposing to query by example (Helén & Virtanen, 2009), imitation or humming (Ghias et al., 1995). As a more practical example, in the Freesound sound sharing platform (Font et al., 2013a), an aggregation of statistics over a large amount of acoustic features is used in order to propose a query by example system where the user can quickly access similar sounds of a previously retrieved one.

Deep learning has been able to produce high-quality audio features that can be re-used for different applications through transfer learning (Choi et al., 2017). These types of approaches make use of pre-trained models as a starting point for different tasks. First layers from trained neural networks often learn similar features which can be applicable for many datasets and tasks (Yosinski et al., 2014). Intermediate layers can serve as higher-level representations which can be used, for instance, in

clustering (Jansen et al., 2017). In this thesis we are interested in identifying what kind of features are adequate for dealing with the wide variety of sounds present in nowadays large and heterogeneous sound collections. Moreover, we investigate approaches to leverage the large amount of publicly available sounds from online collections to learn competitive features without the need of building curated datasets.

1.3.3 Clustering

Search engines enable the users to interact with audio collections by leveraging accompanying metadata. In response to a query, these systems typically return a list of results ranked in order of relevance to the query. The user normally starts at the top of the list and examines one result at a time, until she satisfies her need. For broad or ambiguous queries, the results on different subtopics or meanings will be mixed together in the ranked list. That issue is more severe in collaborative collections, where the crowd-generated nature of the content makes it non-uniformly – and sometimes incorrectly – annotated. It implies that the user has often to go through a large number of irrelevant items to finally locate the one of interest, hence making the task of sound retrieval slow and tedious. One solution for addressing this issue is Search Results Clustering (SRC), which consists of grouping search results into different labeled clusters or categories (Hearst et al., 1995). For instance, clustering search results enables the user to enter a weakly-specified query and explore different topics that have been extracted from the retrieved results. SRC can provide a faster way to retrieve relevant items, enabling topic exploration and alleviating information overlook (Carpineto et al., 2009). Moreover, the ability to rely on machine generated audio features as input of clustering engines can enable to detect perceptually relevant clusters that could have not been discovered from the accompanying crowd-generated metadata.

Nonetheless, the development of clustering engines that can efficiently complement traditional retrieval engines poses some challenges (Carpineto et al., 2009). First, the clustering method should have a decent performance in terms of compactness and separation. In other words, objects

in a cluster should be similar to other objects in the same cluster, and objects from other clusters should be dissimilar to other objects from other clusters (Liu et al., 2010). Then, in order to enable interactive use of the clustered results: meaningful labels have to be assigned to each cluster; the clustering has to be performed online within a short response time; and a graphical user interface should provide an intuitive way to navigate through the clusters. This thesis investigates the use of audio-based SRC for helping users to browse large online sound collections.

1.4 Objectives and outline of the thesis

The main goal of this thesis is to better support exploration and retrieval of sounds in large collaborative collections. We therefore formulate a set of research questions that we address throughout this manuscript:

- (i) How can we make best use of sound collaborative collections in order to build high-quality datasets for supporting advances in machine learning?
- (ii) To what extent and in what way can collaborative collections be directly used to learn audio representations that are useful for classification tasks?
- (iii) How do deep learning features perform for unsupervised classification with a wide variety of sounds?
- (iv) How feasible and valuable is Search Results Clustering for retrieving content from collaborative sound collections?

These questions have led us to the following objectives:

- Build datasets to foster advances in machine listening.
- Investigate and develop novel tools for manual annotation.

- Investigate and develop novel approaches to leverage online audio content in order to learn competitive audio features.
- Investigate which type of audio features are suited for clustering purposes in light of the diversity of sounds present in online collections.
- Integrate clustering into the Freesound search engine by investigating and developing a Search Results Clustering engine. This includes having a fast and effective clustering algorithm, together with an interface that allows users to interact with the clusters.

In order to illustrate the content of this thesis, Figure 1.1 represents the different topics covered. Collaborative Sound Collections are the center component in this thesis, and the main motivation is to contribute to improve their value for creative and research purposes. The arrows imply a contribution from one component to the next. Specifically, we consider Freesound as use case. Based on the Creative Commons philosophy, Freesound¹ is a collaborative platform hosting a repository of Creative Commons licensed audio samples (Font et al., 2013a). At the time of writing, it contains more than 450k sounds and millions registered users. The sounds are very diverse, ranging from field recordings to synthesized effects.

The rest of this manuscript is organized as follows. Chapter 2 describes my perspective on the best way to create high-quality datasets to best promote advances in Machine Learning. In this context, the chapter presents Freesound Annotator, a platform for the collaborative creation of open audio datasets using content from Freesound. As a first result, we describe FSD, a large-scale audio dataset, annotated with categories drawn from the AudioSet Ontology.

Chapter 3 focuses on solving concrete problems induced by common data collection and annotation processes. This is achieved by proposing novel manual annotation tools. Following a user-centered design approach, we propose general-purpose tools for annotating diverse audio

¹<https://freesound.org/>

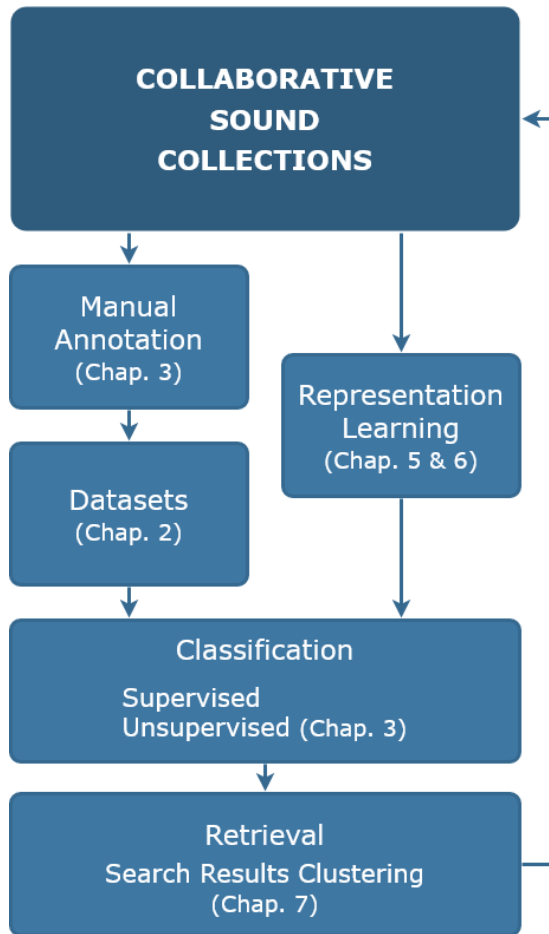


Figure 1.1: Diagram representing the topics covered in this thesis. An arrow indicates that the outgoing element contributes to the incoming element.

content, that supports the exploration and use of predefined categories taken from large hierarchical taxonomies.

Chapter 4 outlines research on identifying which type of audio features are most suited for clustering sounds present in nowadays large and diverse online sound collections. To this end, we compare the performance of several sets of audio features, including state-of-the-art deep learning representations. An evaluation using several diverse datasets is made possible due to the outcomes presented in Chapters 2 and 3.

Chapter 5 and 6 investigates approaches that can leverage user-generated content from online sound sharing platforms, in order to learn audio representations that can serve as a base to various classification tasks. This is achieved by aligning the latent representations learned for audio to those learned for the associated tags.

Chapter 7 discusses the research on Search Results Clustering systems, appropriate for supporting the retrieval of sounds in large collaborative collections. The proposed clustering method relies on a graph-based approach which uses deep audio features as input. We propose a novel way to leverage textual metadata in order to assess the clustering performance at scale. We finally evaluate the SRC system in a real-word context, by observing its use by users in both sound-design and musical-related tasks.

We conclude this thesis in Chapter 8, by summarizing the work done and by providing a discussion of the main findings. Additionally, we highlight perspectives on new methods and research opportunities identified as promising for the organization and retrieval of audio content.

Chapter 2

Data Collection

2.1 Introduction

Digital audio content is nowadays increasingly available over the internet. Many people have contributed to create a large number of audio files from a wide variety of types and audio sources. Much of this data serves creative purposes, often related to the creation of entertainment media, such as movies, music or human-machine interaction. In order to take the most value out of audio collections, their content must be properly stored and organized, facilitating access and retrieval. As introduced in Chapter 1, the growing size of collaborative collections makes storing and organizing their content a challenging task. As a potential solution, machine learning and signal processing techniques have proven successful in the classification of audio content on multiple domains, including (Neumayer et al., 2005b; Herrera-Boyer et al., 2003; Eronen & Klapuri, 2000; Tzanetakis & Cook, 2002; Muller & Ewert, 2010; Hershey et al., 2017).

In this chapter, we highlight the importance and complexity of the different decisions that go in the design and maintenance of large audio collections, in particular with the aim of improving their utility as input for machine learning and signal processing methods. The chapter includes several concrete implementation examples in the context of the Freesound Annotator platform, introduced in Section 2.2.

2.1.1 Annotation and taxonomies

In order to achieve good performance, machine learning and signal processing methods often require large amounts of annotated data, which in its turn relies on manual human annotation as a starting point (Favory et al., 2018). Moreover, the organization of sounds requires the use of a suitable set of predefined categories or classes that cover the types of sounds and/or their characteristics. These sets of categories are referred to as *taxonomies*, and are usually organized as hierarchies.

In limited-scope domains, containing few types of sounds, defining taxonomies is likely to be straightforward. This gets however more complicated as the diversity of the sounds to be described increases. To date,

the largest released public taxonomy for audio contains more than 600 distinct classes (Gemmeke et al., 2017). As a comparison, taxonomies used for image-based classification can often include more than 10k classes, as it is the case for ImageNet (Deng et al., 2009). These large image datasets have contributed to major improvements, e.g. in automatic object identification, and critically depend on dictionaries of concepts taken from large lexical databases such as WordNet (Miller, 1998).

However, the use of lexical databases for audio has not yet been adopted. One possible explanation is that these lexical resources are not always appropriate for describing audio content. They contain concepts that are either too specific for conveying audio description, or either not specific enough. This can be understood by the fact that sounds are often more abstract and ambiguous than images and, as a consequence, the necessary vocabulary can include words with different meanings. As an example, *echo* and *reverberation* are considered different concepts in audio, although they originate from the same acoustic phenomenon. However, according to the lexical resource WordNet, the two concepts are considered synonyms¹. Furthermore, there are many concepts in music which are simply not present in lexical resources. For example, the concept of *jingle*, defined as “a short song or tune used in advertising and for other commercial uses”², does not exist in WordNet. And there are numerous highly-specific concepts used by music producers for describing the content they use or create which are also not included in general lexical databases. The first challenge in generating high-quality annotations therefore lies in deciding upon the set of concepts and categories to describe the sounds, i.e. the selecting a suitable taxonomy.

2.1.2 Datasets

Datasets of sounds have been proposed for many different purposes, including classification or auto-tagging, speaker identification, source separation, and music transcription. Since this thesis focuses on the im-

¹<http://wordnetweb.princeton.edu/perl/webwn?s=echo>

²<https://en.wikipedia.org/wiki/Jingle>

provement of classification and tagging of audio content, we will focus on datasets designed for this purpose. Our interest lies mainly on the annotation of content with semantic labels, which can be used to pinpoint the audio source(s) present in a file, describe properties of said source(s), identify the acoustic scene, and so on. The main goal is to contribute to advances in machine listening, with the aim of obtaining maximum value from existing sound collections. This entails proposing and improving methodologies for the development of datasets. For the remaining of this manuscript, we consider a dataset as being composed of (i) a sound collection i.e. a set of audio files, (ii) a set of pre-defined categories and, (iii) the annotations that associate the audio content to the categories.

2.1.3 Dataset desiderata

We consider the collaborative creation of audio datasets by a community of users. Embracing the ideas described by (McFee et al., 2016) and (Stodden et al., 2016) for sustainable information retrieval evaluation and reproducibility of computational methods, the datasets and their creation should ideally subscribe to the following principles:

- **Size.** Having sufficient data is one of the major concerns for many researchers. Small datasets may limit the application of certain machine learning techniques, where larger datasets are needed. In particular, novel deep learning approaches often require large amounts of data.
- **Openness.** It is indispensable that datasets are completely open, including audio and ground truth data. Both should be available under open licenses that allow the free distribution and reuse. Furthermore, other relevant data generated during the dataset creation process could be made available (e.g., annotation procedures and the original raw annotations). Keeping this information as open as possible aids in the detection of potential issues or biases in the collection process.
- **Transparency.** It is important that workflows in the dataset creation process are clear to the users. This will allow a better understanding of the dataset itself, its potential and limitations. In this regard, facilitat-

ing the exploration of the content through intuitive interfaces is a crucial functionality that is often overlooked. Moreover, splits of datasets (e.g., train and test) should be proposed and made publicly available for system benchmarking and to ensure reproducibility, so that researchers can carry out experiments with directly comparable results.

- **Diversity.** Many datasets focus on a specific type of sounds rather than covering a broad spectrum of sound types. Diverse datasets are more suited for covering the entire range of sounds present in online collections, such as Freesound, and are therefore preferred. Moreover, machine listening systems can be used in various distinct applications and devices, e.g. to monitor sounds in cities (Bello et al., 2019), healthcare (Hüwel et al., 2020), bioacoustics monitoring (Xu et al., 2017), surveillance (Crocco et al., 2016), smart devices (Do et al., 2018), or conversational agents (Park et al., 2020). Some online collections are collaboratively generated by users with different backgrounds and levels of expertise, and using different devices for recording or producing sounds. By reflecting the variability in input and recording conditions, the resulting content can better serve a variety of aims.

- **Dynamism.** It is desirable, and even necessary, that the dataset and its collection be the subject of constant discussion and improvement. Indeed, research has been devoted to the analysis and critique of existing datasets (Sturm, 2013), as well as proposing updated versions (Kereliuk et al., 2015). We envision such criticism and proposals happening in a collaborative online platform where detected faults and issues can be discussed and adequately addressed. This would imprint a dynamic character to datasets which could be versioned and updated with contributions from the community.

- **Sustainability.** To ensure that datasets with the aforementioned properties stand in the long term, a sustainable approach is required not only for gathering audio content and annotations, but also in its maintenance. In the ideal scenario, the community acts as a continuous source of information at different levels. Ideally, the community would be self-sufficient, providing a large-scale source of audio-related content. As a

matter of fact, previous works have adopted similar ideas for gathering large amounts of user-provided data. Notable examples are AudioSet, based on YouTube videos (Gemmeke et al., 2017), and ImageNet, based on Flickr and other search engines (Deng et al., 2009). In order to construct the corresponding ground truth annotations at a large scale, a substantial part of the annotations is likely obtained through crowdsourcing. Finally, technical maintenance requirements should be kept as low as possible.

The remaining of this chapter introduces the Freesound Annotator platform, our effort towards sustainable construction and maintenance of collaborative datasets with the aforementioned properties.

2.2 The Freesound Annotator platform

Freesound Annotator is a website that allows the collaborative creation and curation of open audio datasets. It serves three main goals in facilitating i) the management of datasets, ii) the creation and verification of annotations, and iii) browsing and exploration by users. Currently, Freesound Annotator hosts the FSD dataset, introduced in Section 2.2.1.

Freesound Annotator was originally released in April 2017, but its development is an ongoing process. Initially, it started by providing tools for exploring taxonomies and validating automatically-generated annotations. These tools were used by users early-on in the development process in a controlled scenario, in order to start gathering annotations and collecting feedback for improving the platform. Additional features were subsequently added e.g. for providing clearer guidelines, better category and taxonomy visualisations, and practical answers to common doubts. Since one of our intentions is to crowdsource the annotation of content, we also implemented several quality control mechanisms (Ipeirotis et al., 2010; Sabou et al., 2014). One of such mechanisms consists in prompting the annotating users with test cases, i.e. audio clips with known annotations. The users are not aware of this fact, which allows us to assess the reliability of their answers and filter potential spam. Another indicator of

annotation quality is the agreement between multiple users for the same category and clip, which is therefore another key mechanism for generating ground truth annotations. The implementation of these features was crucial when advertising our platform, and allowed us to collect contributions from more than 500 people. As a complement to crowdsourcing, we also developed specialized tools suited for expert annotators, needed to overcome more challenging annotation tasks (Chapter 3). Finally, monitoring tools were added, allowing us to follow the current state of the dataset by visualizing the progress for each category’s annotation. Moreover, the monitoring tools also enabled us to easily inspect users’ contributions or quickly detect mistakes.

As a technical remark, the platform uses the Django Python Framework for the development of the application and relies on Postgres as a database. The database model allows us to store, manage and retrieve sounds and their annotations. Moreover, since it is based on crowdsourcing, a user model allows us to keep track of the contributions of the users. Maintaining these models allows for instance to assign some scores to annotations in order to prioritize certain types of sound and to reach users’ agreement faster on the validation of annotations by multiple people. We also store and keep all the contributions made by any user so that they can be reused in any other ways. The code of the platform is available publicly at: <https://github.com/MTG/freesound-datasets>

2.2.1 FSD: a dataset for general machine listening

Hosted on Freesound Annotator, the FSD dataset is a collection of audio samples from Freesound. As described by Fonseca et al. (2017b), each clip in FSD is annotated with categories drawn from the AudioSet Ontology, the largest publicly-available taxonomy for audio. It is a hierarchical collection of over 600 classes. Currently, FSD displays annotations that express the presence of a sound category in an audio sample. We decided to first focus on the FSD dataset, as it can serve many applications in general machine listening. The creation of FSD started by automatically populating each category of the AudioSet Ontology with several candi-

date audio samples from Freesound. This process generated over 600k candidate annotations. In the following, we introduce functionalities of Freesound Annotator, in its application to the FSD dataset.

2.2.2 Functionalities of the platform

To verify the validity of these automatically generated annotations, we developed a validation tool with an interface that helps users to understand a category and its context in the AudioSet Ontology.

- **Validation of candidate annotations.** This validation tool is deployed in the Freesound Annotator platform. Figure 2.1 shows the part of the interface used by the users in order to familiarise themselves with a given category. It displays information such as the name, description, sibling and children categories of a specific category.

The screenshot shows a web interface for familiarizing users with the 'Sawing' category. At the top, there is a header 'Familiarize yourself with' followed by a 'Sawing' button and a 'Choose another category' button with a refresh icon. Below this is a table-like structure with the following sections:

- Hierarchy:** Sounds of things > Tools > Sawing
- Description:** Sounds of a tool consisting of a tough blade, wire, or chain with a hard toothed edge, used to cut through material, most often wood. Includes both manual and motorized saws.
- URI:** <http://en.wikipedia.org/wiki/Saw>
- Examples:** Two audio player thumbnails showing waveform and spectrogram views.
- Siblings:** Hammer, Filing (rasp), Sanding, Power tool, Jackhammer
- FAQ:** Are chainsaw sounds considered 'Sawing' sounds?

Figure 2.1: Screenshot of the familiarisation interface of the Freesound Datasets platform validation task

After getting familiarized with a given category, the user is presented with a page containing several sounds for the which he is asked to validate

the presence of the sound category (see Figure 2.2). At first, after submitting his contribution for the page of 12 sounds, the user was thanked and was asked if he wanted to continue or select another category. We realized that many sporadic users were then only contributing with one page, which was not very profitable. As already identified in crowdsourcing annotation scenario (Sabou et al., 2014), users tend to produce better annotations after having spend some time annotating, often referred as a training phase. Therefore, we decided to organize the validation task in sessions of 6 pages, before thanking the user and proposing him some deserved rest. As a result, we saw an increase of the number of contributions from some users, which tended to provide much more annotations.

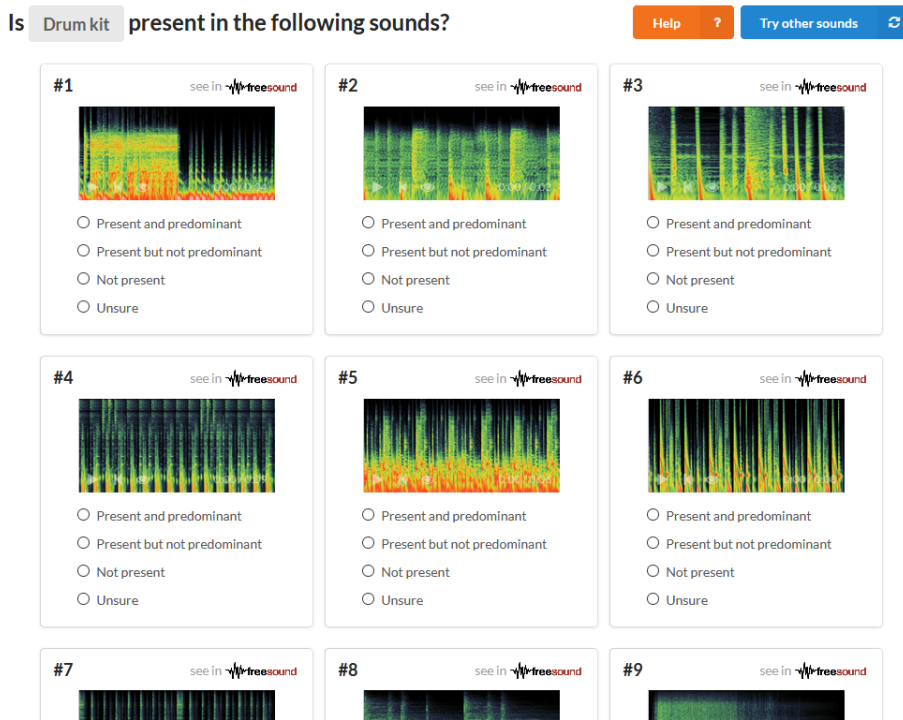


Figure 2.2: Screenshot of the interface when performing the validation task.

• **Beginner and advanced tasks.** The Audioset Ontology is composed of many categories. Some are fairly easy to identify for any human being (e.g. dog bark), whereas others can be more challenging (e.g. soprano saxophone). We therefore decided to divide the validation task in two. One called *beginners* that would assign a random category among a few simple ones, and another one called *advanced*, that allows the user to choose a category among more complicated ones. In order to be able to contribute to the *advanced* task, users were first asked to contribute at least once to the *beginners* task. This would for instance mitigate the effect of user spam for any category or enforce the idea of progression through a gamification rewarding concept (Morschheuser et al., 2016).

• **Gathering participants.** Few design decisions were made related to gamification. Nonetheless, we provided a ranking of the top contributors, displaying all time top contributors as well as last 24 hours top contributors, as shown in Figure 2.3. This was mainly used during several *tagathon* events, to motivate the gathered participants in their annotation process.

• **Monitoring tools.** In order to follow the progression towards the annotation of the collection, we provided tools in order to monitor different aspects of the platform. These include: the number of contributions, ground truth annotations, repeated wrong answers to quality control test cases, repeated annotator disagreement on validation, and so on. Moreover, individual contributions of annotators can also be inspected.

• **Amend mistakes.** In order to facilitate the detection and correction of mistakes in dataset releases, we provide a way to explore and report errors in existing labels as shown in Figure 2.4. These error reports are kept in database for allowing further inspection by maintainers of the dataset.

• **Gather more candidate annotations.** The creation of the dataset started with the automatic generation of candidate annotations, that were obtained using a tag-matching process. As a result, some categories did not have a large amount of candidates. New tag-matching rules can be modified from the web application, in order to facilitate the generation of

All time top contributors			Last 24h top contributors		
#	Username	Number of contributions	#	Username	Number of contributions
1	★ XavierFav	4139	1	★ bvlbhor	2041
2	★ idrojsnop	3734	2	★ minzwon	1284
3	★ minzwon	2439	3	★ floaiciga	1137
4	albincoreya	2356	4	XavierFav	767
5	helenacm	2343	5	idrojsnop	718
6	emirdemirel	2341	6	Nerkamitilia	648
7	mmiron	2254	7	helenacm	572
8	hsercanatli	2202	8	albincoreya	553
9	floaiciga	2121	9	emirdemirel	473
10	bvlbhor	2041	10	gil_dori	456
11	xserra	1979	11	pc2752	372
12	gil_dori	1684	12	sertansenturk	331
13	txirimiri	1407	13	RafaelCaro	300
14	voices_pond	1407	14	txirimiri	252
15	dole25	1372	15	sergio.oramas	144

Figure 2.3: Screenshot of the annotator ranking during an annotation event.

new candidate annotations. The form shown in Figure 2.5 allows maintainer to visualize existing mapping rules, or directly adding sounds from Freesound using their numerical identifiers, or propose new tag-matching rules. Once submitted, boolean queries are performed to the database, and some statistics are presented to the user. Figure 2.6 shows the number of sound retrieved and the proportion that are already considered as candidate in the platform. Additionally, the proportion of existing validation votes are displayed, as well as existing tags for the sounds reported as not containing the specific sound category through the validation task. Finally, random examples are shown at the bottom of the page, and the user can quickly assess their validity, in order to get an estimate of the quality of the new mapping.

FSD50K → Electric guitar

Hierarchy	🌳 > Music > Musical instrument > Plucked string instrument > Guitar > Electric guitar
Description	Sounds of a guitar that uses a pickup to convert the vibration of its strings (which are typically made of metal, and which occurs when a guitarist strums, plucks, or fingerpicks the strings) into electrical impulses, which are then amplified and converted to sound.
URI	http://en.wikipedia.org/wiki/Electric_guitar
Siblings	Strum Tapping (guitar technique) Bass guitar Steel guitar, slide guitar Acoustic guitar
# audio samples	687

Audio samples from Electric guitar

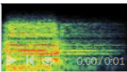

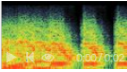

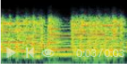

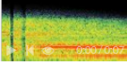

Sound	Freesound URL	Partition	Report error
	see in freesound	dev	 reported by 0 user
	see in freesound	dev	 reported by 0 user
	see in freesound	dev	 reported by 0 user
	see in freesound	dev	 reported by 0 user

Figure 2.4: Screenshot of a category exploration table, with the error report tracking.

Monitor the mapping of A capella

Mapping rules originally used:
Include tags [acapella] **Omit tags**

Mapping rules used in the platform:
-

Filter by ID

Comma separated Freesound IDs

Freesound IDs string

+ Add new candidates

Filter by tags
Add the tags you want to be selected

Include tags **Omit tags**

Comma separated tags (e.g.: dog, bark, panting)

Comma separated tags (e.g.: dog, bark, panting)

instrument

Stem tags

Acapella ✖

+ Add alternative tag group

Stem tags

Submit

Figure 2.5: Screenshot a section of the mapping rule monitoring page which allow maintainers to retrieve new candidate annotations.



Figure 2.6: Screenshot of inspection section of the mapping rule monitoring page for a specific category and submitted mapping.

2.3 Outcomes and conclusion

Through the existence of the Freesound Annotator platform, several dataset releases have been made accessible for the public for different purposes:

- **FSD50K** is the latest and main release of FSD (Fonseca et al., 2020a). It contains 51,197 audio clips from Freesound, totalling over 100h of content manually labeled using 200 classes drawn from the AudioSet Ontology. The audio clips are licensed under Creative Commons license. The annotations are provided at a clip level, meaning that they only convey the presence of a sound category in an entire clip whether than providing exact location in the clips. To our knowledge, it is the largest fully-open dataset of human-labeled sound events ever released. In total there are 152,867 annotations in the dataset. In order to enable reproducibility and fair comparability results when using the dataset, it provides pre-computed splits. The development set consists of a training and a validation part. It contains labels that are correct but could be occasionally incomplete. The evaluation set has been annotated exhaustively, meaning that all labels are correct and complete for the considered vocabulary.

- **Collection of datasets for clustering validation.** This is a collection of relatively small datasets that have been created using data from FSD. They aim at enabling an evaluation of different audio features for the clustering of diverse types of sounds. This collection contains 44 datasets organized in 6 families comprising a total around 30k sounds. These datasets will be presented more in detail in Chapter 4, as long with a comparative study of feature performances for the unsupervised classification of sounds.

- **FSDKaggle2018** (Fonseca et al., 2018). Dataset containing 11k audio clips and 18 hours of training data unequally distributed in 41 classes of the AudioSet Ontology. It was collected for the DCASE Challenge 2018 Task 2³, which was run as the Kaggle competition Freesound General-

³<http://dcase.community/challenge2018/task-general-purpose-audio-tagging-results>

Purpose Audio Tagging Challenge⁴.

- **FSDnoisy18k** (Fonseca et al., 2019a). Dataset collected with the aim of fostering the investigation of label noise in sound event classification. It contains 42.5 hours of audio across 20 sound classes, including a small amount of manually-labeled data and a larger quantity of real-world noisy data. Described in its companion site and in our ICASSP 2019 paper.

- **Free Universal Sound Separation Dataset** (Wisdom et al., 2020). This dataset contains arbitrary sound mixtures and source-level references, for use in experiments on arbitrary sound separation. It uses a subset from FSD50K and was used during the DCASE2020 Challenge Task 4⁵.

Additionally, there exist other datasets that are partially built using content from FSD:

- **FSDKaggle2019** (Fonseca et al., 2019b). This dataset contains data from FSD and from the Yahoo Flickr Creative Commons 100M dataset (YFCC). It includes 29,266 audio files annotated with 80 labels from the AudioSet Ontology. It has been used for the DCASE Challenge 2019 Task 2⁶, which was run as a Kaggle competition titled Freesound Audio Tagging 2019⁷.

- **DCASE 2019 task 4** (Turpault et al., 2019a) . The dataset is composed audio clips recorded in domestic environment or synthesized to simulate a domestic environment. It was used in the DCASE 2019 task 4 challenge⁸. A subset of FSD is used as foreground sound events for the synthetic subset of the dataset.

⁴<https://www.kaggle.com/c/freesound-audio-tagging>

⁵<http://dcase.community/challenge2020/task-sound-event-detection-and-separation-in-domestic-environments-results>

⁶<http://dcase.community/challenge2019/task-audio-tagging>

⁷<https://www.kaggle.com/c/freesound-audio-tagging-2019>

⁸<http://dcase.community/challenge2019/task-sound-event-detection-in-domestic-environments>

Chapter 3

Improving the Manual Annotation of Audio Content

In the previous chapter we introduced FSD, a large-scale open audio dataset based on Freesound content and annotated with categories drawn from the AudioSet Ontology. The data collection process started by automatically generating labels and verifying their validity using a manual validation tool. This approach produced a considerable number of annotations, that have already proved useful in the machine-listening community (Fonseca et al., 2018). However, generating annotations automatically presents a number of shortcomings such as generating incorrect or non specific labels and failing to generate existing ones.

In this chapter, we present novel annotation tools to solve specific problems induced by our initial data collection and annotation processes. In addition, these annotation tools were designed in order to benefit both: i) the process of publishing audio content in an online platform (e.g., when content creators upload content to Freesound), and ii) post-annotation steps, in which users of a platform can collaboratively contribute to the annotation of content and ground truth generation (e.g. in the Freesound Annotator platform). Assigning labels from a large vocabulary to audio resources is a difficult task for non-experts and creates a number of challenges. We try to mitigate these challenges by proposing interfaces that guide users in the process. Following a user-centered design approach, we propose general-purpose annotation tools for annotating diverse audio content. We take advantage of the AudioSet Ontology which provides a hierarchical taxonomy of broad acoustic categories. The main goal is to facilitate the exploration and use of predefined categories taken from large taxonomies.

3.1 Introduction

Recent advancements in machine learning partially come from the popularity of online sharing platforms, which made large amounts of data available (Russakovsky et al., 2015). In these platforms, description and tagging systems have become increasingly popular. Users can add textual descriptions or keywords (i.e., tags) to Internet resources (e.g., web pages,

images, music) without relying on a controlled vocabulary. This makes it less demanding for users than, for example, classifying objects into predefined categories. Although these user-generated descriptions enable the development of valuable search tools for online-shared content (Marlow et al., 2006), they are not always directly adequate for the effective management of multimedia content. Indeed, the interoperability of the content descriptions is fundamental for information sharing, exchange and reuse. It is therefore crucial to have semantic content metadata that is understandable and processable both by machines and humans. One of the challenges in making use of shared audio content comes from the fact that it is provided by various sources and by authors with different backgrounds and levels of expertise. Therefore, the content is often unstructured and not properly annotated, which hinders its efficient retrieval. Moreover, there is a scarcity of tools and established methods to aid users in the task of annotating audio content through common procedures. Guiding users intelligently throughout the annotation process would allow a reliable, uniform and complete description of the content, thus facilitating its sharing.

To address this issue, taxonomies allow to organize and structure concepts. In the audio-related fields they are the first step towards the classification of sounds into groups based on different subjective or contextual properties (Schafer, 1993). Disparate taxonomies have been developed based on subjective similarity, sound source or common environmental context. However, since sounds are multimodal, multicultural and multifaceted, there is not a common taxonomy that allows to organize large and diverse sound collections. Some works proposed taxonomies for environmental sounds, based on the interaction of materials (Gaver, 1993) or according to their physical characteristics (Schafer, 1993). More recent research on studying soundscapes shows that the taxonomical categorisation of environmental sounds is not trivial and involves many different fields, e.g., human perception or urban design (Brown et al., 2011; Salamon et al., 2014). For musical content, many music genre taxonomies appeared from the Music Industry and its consumers. Yet no standard taxonomy has been established since it depends highly on our cultural

contexts. In fact, each distributor has his own strategy towards its targeted market (Pachet & Cazaly, 2000).

Despite all the accomplishment in designing specific taxonomies, the creation of larger, general-purpose taxonomies has recently gained attention among the research community (Gemmeke et al., 2017). Instead of focusing on the recognition of a specific subset of sounds, general-purpose taxonomies enable tasks that aim to recognise and describe a wider (and usually more generic) range of sounds (Fonseca et al., 2018). Methods to solve these tasks are desirable, for example, in environments such as smart buildings or smart cities and more generally in IoT applications. Another application is the automatic description of multimedia content in the context of large online collections like Freesound (Font et al., 2013a) or Youtube. This can enable the enhanced organisation and retrieval of multimedia content, thus making it more accessible to the public. In these cases, training general-purpose systems with large-vocabulary audio datasets seems more suitable to be able to describe a wide variety of content types.

The recently released AudioSet Ontology proposes one of the biggest taxonomies which structures 632 audio-related categories (Gemmeke et al., 2017). Rather than being domain-specific, it contains the most common concepts used for describing everyday sounds. AudioSet has a companion website that includes a web-interface to navigate through the taxonomy and listen to sound examples, which provides an overview of its content¹). Sounds related to a wide variety of concepts, such as nature, urban design, music and culture. Consequently, sound-related taxonomies may evolve and adapt, and it is important for people to understand, use and discuss about them. For this reason, proposing tools and interfaces for browsing taxonomies would lead to vast advancements in the many related fields. Likewise, these tools can assist the annotation of the content in online sharing platforms, which would facilitate its use for research or multimedia sharing.

The rest of the chapter is as follows. In Section 3.2, we first explain the motivations of this work. Section 3.3 describes the annotation tools

¹<https://research.google.com/audioset/>

we developed. An evaluation with users is presented in the following section and we finally conclude the chapter in the last section.

3.2 Motivations

3.2.1 Audioset

AudioSet is the largest dataset of sound events ever released, consisting of more than 2M audio clips manually labeled using 527 classes drawn from the AudioSet Ontology (Gemmeke et al., 2017). Contrary to many existing datasets, it puts emphasis on general-purpose sound event recognition, enabling the description using a wide variety of sound classes. The ontology contains 632 categories organized in a hierarchy. It consists of a hierarchical structure that has a maximum depth of 6 levels, starting with broad and common concepts (e.g. Music and Speech), and up to much more specific and scarce categories (e.g. Bicycle bell). This hierarchy facilitates the exploration of the classes by introducing them in a gradual level of specificity, which can facilitate the annotation process. The creation of the ontology started from an analysis of web text using natural language processing techniques that led to a list of terms. These terms were then manually organized into a hierarchy, which was then compared to existing taxonomies, and adapted consequently. The ontology then received further modification after using it for annotation purposes, until it was eventually finalized and published. It contains many information about the classes, such as name, description, and sometimes URIs from WordNet (Miller, 1998) or Wikipedia.

Although the hierarchy provides a way to explore the ontology, the large number of categories makes it quite difficult to visualize and remember the entire hierarchy. As an example, Figure 3.1 represents slightly more than 10% of the entire ontology.

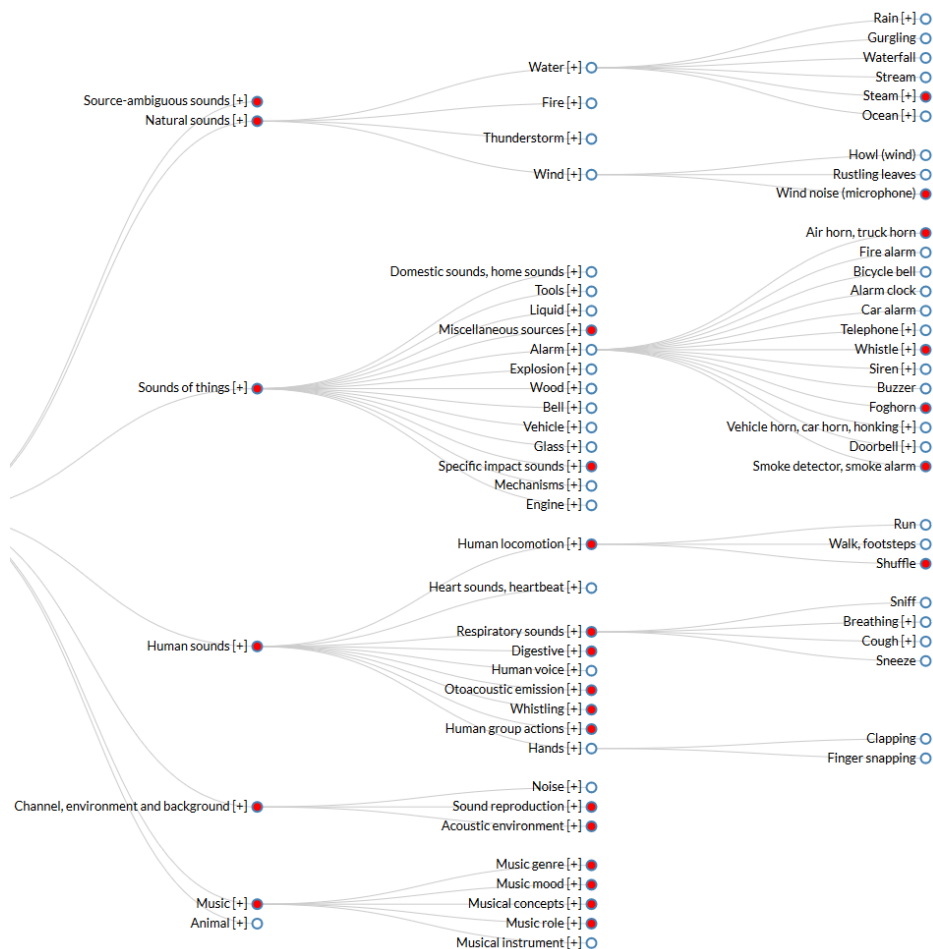


Figure 3.1: Representation of a small part of the AudioSet Ontology hierarchy.

3.2.2 Motivating new annotation tools

One of our goals is to provide annotation tools that can be suited for the post-process annotation of content in a platform such as Freesound Annotator. In this context, the annotations that were automatically generated

presented a number of shortcomings. For instance, an automatic process can generate incorrect or not specific labels, and it can also fail to generate some labels. We argue that the usefulness and reliability of datasets increase with the proximity of its annotations towards what we denote as *complete* or *exhaustive* labeling (i.e., all the acoustic material present in the audio file is annotated).

To achieve this complete labeling status through manual annotation, a number of actions would be required. First, assuming the existence of automatically generated annotations, it would be needed to validate them. Then, missing labels should be *generated*, and generic or unspecific labels should be further *refined*. Additionally, the refining scenario that we will present can also benefit users that upload content, where the iterative specification of labels from generic ones can ease the annotation process. The two annotation tools presented in the next section address the two latter issues.

3.3 The annotation tools

In this section we describe the two novel interfaces that we developed. The code is available at: <https://github.com/MTG/freesound-datasets/tree/annotation-tools-FRUCT2018/>. Both tools are implemented mostly with web client languages, which allows their easy integration in other projects. The Audio Commons Manual Annotator (AC Manual Annotator) aims at adding missing labels, whereas the Audio Commons Refinement Annotator (AC Refinement Annotator) allows to refine and specify existing labels. These tools can be useful not only to annotate during a post-processing stage, like in Freesound Datasets, but also to provide annotations when a user publishes content in an online platform such as Freesound. Both of the tools focus on annotating a single sound resource at a time. The audio content is accessible from a player displaying the spectrogram of the sound, which can facilitate the localisation and recognition of sound events in the clip (Fig. 3.2 & 3.4) (Cartwright et al., 2017).

3.3.1 Generate annotations

With the AC Manual Annotator, labels can be assigned to an audio clip. The main idea behind this interface is to provide a way to facilitate the quick overview of categories. Moreover, considering the large size of the hierarchical structure in taxonomies like AudioSet, it is important to show the location and context of the categories within the hierarchy. Another design criteria was to allow the comparison of different categories by simultaneously displaying their information. In the proposed interface, a text-based search allows to locate categories in the taxonomy table. We used text from the category names and descriptions to perform some trigram-based queries (a feature that Postgres, our database backend, implements²). The taxonomy table allows users to open parts of the taxonomy in order to visualise children categories simultaneously. For each category, textual descriptions are shown, along with sound examples when available (Fig. 3.3).

A typical use workflow would consist in:

- Listen to the sound sample (Fig. 3.2, top).
- Use the text-based search to locate categories in the taxonomy table (Fig. 3.2).
- Explore the taxonomy table to understand well the located category, and perhaps find other more relevant categories (Fig. 3.3).

3.3.2 Refine annotations

The AC Refinement Annotator displays some previously existing labels as rows, as it can be seen in Fig. 3.4. The annotator can examine their location in the AudioSet hierarchy as well as their siblings and children categories. By making use of the hierarchy, the main goal of this tool is to aid the annotation process by providing an iterative way of specifying

²<https://www.postgresql.org/docs/9.6/static/pgtrgm.html>

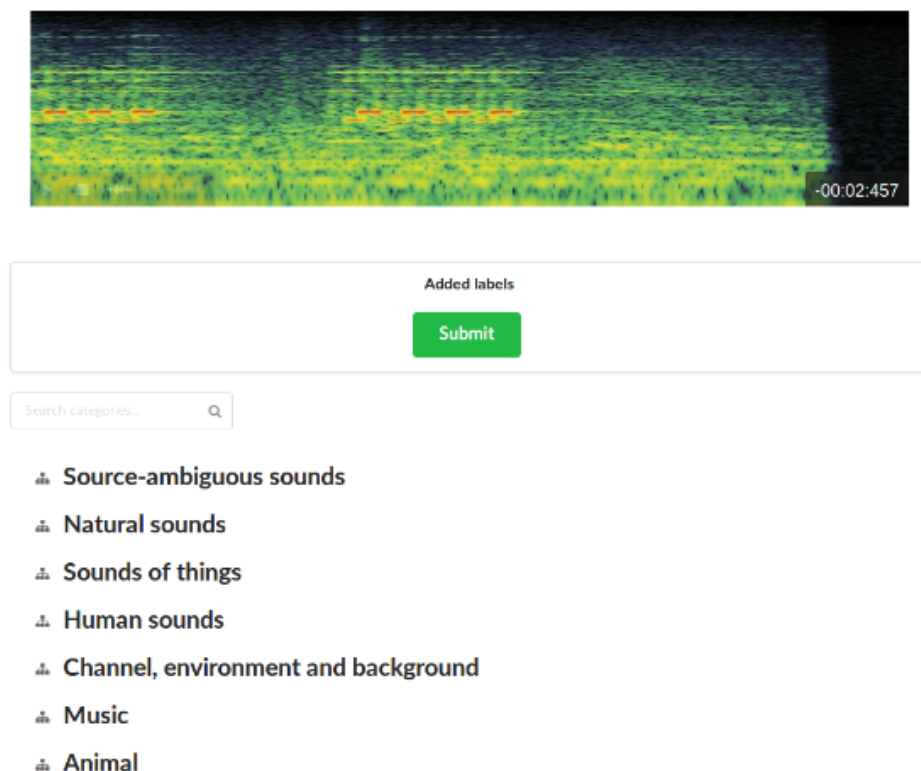


Figure 3.2: Screenshot of the Audio Commons Manual Annotator

the type or nature of the content. Fig. 3.5 shows how the children categories of the proposed label “Guitar” are displayed in a dropdown, which allows to modify the label and define it more precisely. For every label, popups show the category description and examples when available (Fig. 3.6). Moreover, it is possible to duplicate a label using the icon at the top right corner of a label path. This allows, for instance, to specify a label by adding two of its children categories. In the final step of the refinement process, the user is asked to verify the *presenceness* of the selected category in the audio clip.

A typical use workflow would consist in:

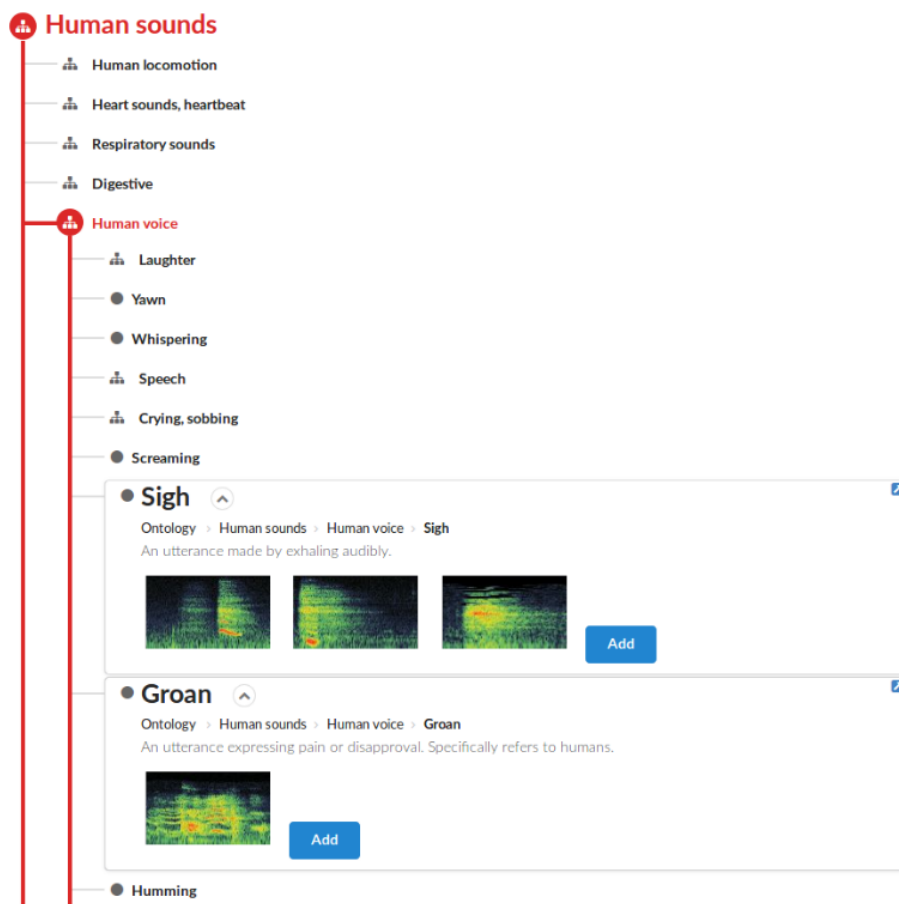


Figure 3.3: Screenshot of the Audio Commons Manual Annotator taxonomy table, showing the descriptions and examples of “Sigh” and “Groan”, together with their hierarchy location

- Listen to the sound sample (Fig. 3.4, top).
- Inspect the proposed labels (Fig. 3.4).
- Refine the proposed labels by inspecting the related siblings and children (Fig. 3.5 & 3.6).

- Validate the presence of the proposed or refined category.

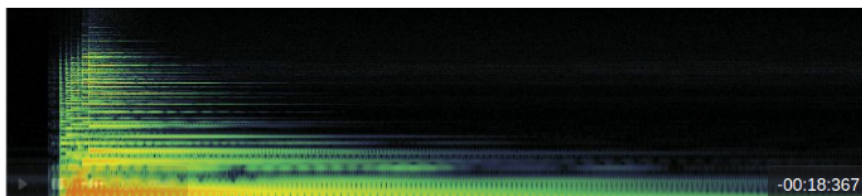
The screenshot displays three sequential panels of the annotation interface. Each panel shows a breadcrumb trail of categories and a list of radio button options. The first panel shows the path 'Music > Musical concepts > Chord' with options: 'Present and predominant', 'Present but not predominant', 'Not present', and 'Unsure'. The second panel shows the path 'Music > Music genre > Blues' with the same four options. The third panel shows the path 'Music > Musical instrument > Plucked string instrument > Guitar' with a 'Select' button and the option 'Present and predominant'.

Figure 3.4: Screenshot of the Audio Commons Refinement Annotator displaying a sound sample and its three suggested label paths

3.4 Evaluation

In the context of sound collections annotation, there is a need for proposing new manual interfaces to properly annotate audio content, with labels that are comparable and of the same nature. In this experiment, we present our user-centered design process on the development of novel tools for

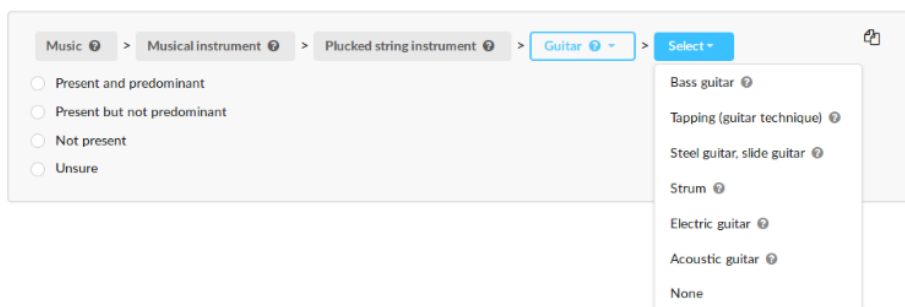


Figure 3.5: Screenshot of the Audio Commons Refinement Annotator showing a dropdown displaying the children categories of “Guitar”

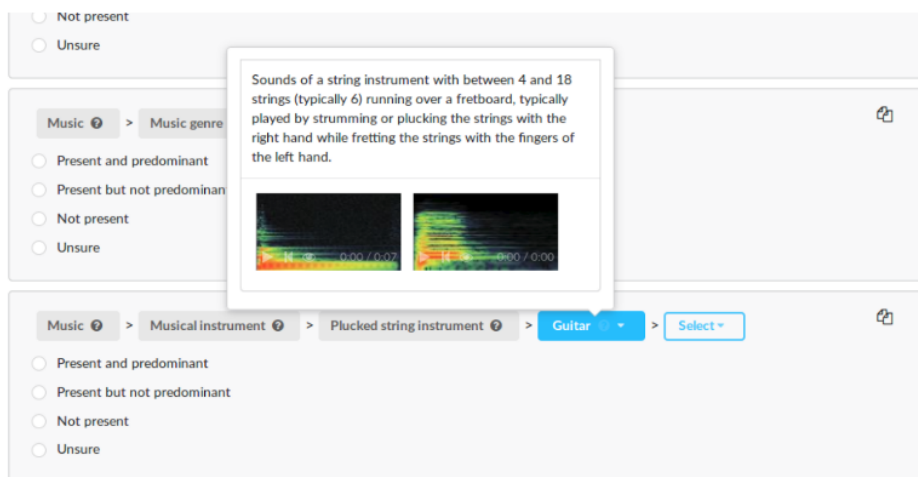


Figure 3.6: Screenshot of the Audio Commons Refinement Annotator showing the description and examples of the “Guitar” category in a popup

annotating audio content from a wide variety of types. We use the annotator tools as *technology probes* to observe their use in a real context, to evaluate their functionalities and to inspire new ideas (Hutchinson et al., 2003).

3.4.1 Methodology

We gathered eight participants with different levels of expertise. Each one of them was provided with one of the tools and was asked to annotate a list of sounds one by one. We selected sounds from the Freesound Annotator platform featuring one or more of the following aspects: (i) containing multiple sound sources, (ii) presenting background noise or (iii) being hard to recognise. This process resulted in a list of 9 and 15 sounds for the generation and refinement tools respectively. Some guidelines were shown to the participants, together with verbal explanations given by the examiner. At the end of the task, they were provided with a questionnaire containing some usability and engagement questions. Finally, semi-structured interviews were carried out, including open-ended questions as well as specific questions related to observed behaviors during the development of the task. This enables discussion using thematic analysis in order to identify emerging themes from participants' answers.

3.4.2 Results and discussion

Finding a category in the taxonomy

It is essential to provide ways for efficiently browsing and exploring such an extensive set of audio categories. Text-based search provides a way for people to find categories with their own words. This is particularly efficient when the annotator recognises the sources and want to quickly add the corresponding audio category to the content. As a way to improve the retrieval from the text-based search, one participant proposed to add some of the children of the retrieved categories to the results. This option was tested when developing the search engine, but was discarded because it tended to add a lot of results which made the localisation of the relevant categories harder. Moreover, we could also use external lexical resources such as WordNet or Wikipedia to improve system's recall, by using synonyms terms and page content terms respectively.

However, text-based search can fail when the annotator is not familiar with the vocabulary. She can then rely on the hierarchical structure of

the categories. Tree visualisations are a direct representation of it, and can help by allowing to iteratively define more precise concepts starting from the broader upper levels of the taxonomy. As well, tables are a natural way for browsing collections of items. The taxonomy table we provided in the AC Manual Annotator aims at combining tree and table structures in order to allow efficient and fast exploration of the categories. Moreover, locating similar categories close from each other helps to refine and validate the choice of a category (especially for categories that are almost identical and differ only in small details).

Exploring the taxonomy

The hierarchy structuring the audio related concepts assumes that deep located categories convey more information than the others. Therefore, it is important to use labels as specific as possible in order to accurately describe the audio content. When using the AC Refinement Annotator, some participants showed interest in seeing all the hierarchy at once. However, we believe that the task is facilitated if only the relevant context for each step of the iterative process is shown. Specifying labels in an iterative fashion (i.e., progressively, such that their meaning is narrowed down in every step) seems to be helpful. It can ease and speed up the generation of accurate labels by focusing on the most relevant semantic audio aspects. Nonetheless, during the navigation through the different levels of specificity in the hierarchy, a participant was sometimes not inspecting, or hesitating to check, the children of a category. This occurred due to several reasons: (i) since no sound examples were available in the present category, he assumed this would also be the case in deeper hierarchy levels. Hence, he decided not to explore this branch due to lack of confidence with it; (ii) he also assumed that since the original category was not appropriate, none of the children would be either (where in fact, one of them was). The AC Manual Annotator mitigates this problem and facilitates quick inspection of the categories, since the children can be automatically displayed when a category is selected in the taxonomy tree.

Difficulty in recognising a sound identity

In the context of post-processing annotations of audio content, the annotator is typically not the publisher of the content. Hence, the annotator usually does not know the details of the recording conditions or what sound sources were captured. Furthermore, listening to the sound does not necessarily lead to the identification of the sound source(s) as it can sometimes be a very complex task. Under these circumstances, for the audio content that annotators were not able to recognize, the following behaviors were observed. When using the AC Manual Annotator, the annotators tended to choose abstract categories that do not convey the source identity, but rather some other aspects of the sound source (e.g., onomatopoeic labels that phonetically imitate, resemble, or suggest the sound it describes). In the AC Refinement Annotator tool, where participants were guided towards the identification and specification of the sources, they usually stopped at a certain hierarchical level, thus providing some imprecise labels. As expected, labels gathered with the *generation* tool were much more different than those gathered with the *refinement* tool. One of the reason was that with the AC Manual Annotator tool, users chose different abstract labels for describing the content, since their exact meaning seems to vary across annotators.

To improve the consistency of the produced labels, it was discussed to give access to the metadata that often accompany online shared media, e.g., title, description and tags. These informations can guide annotators on understanding the context and providing more accurate annotations. However, some participants argued that these informations should not be given at first. For them, access to metadata should be an additional aid that could be requested only after having spent a certain effort on analysing the audio content. Providing directly the metadata would correspond more to a transcription task, where annotators could focus only on the metadata, and forget some important sound aspects that the metadata fail to convey.

The annotators' commitment is highly variable

In addition to the precision of labels, the AC Refinement Annotator also allows to explore siblings categories that can sometimes correspond to slightly different concepts. This enables correcting the, potentially noisy, automatically generated labels. However, this feature led to variable results in terms of labels produced and time spent annotating. Users of the *refinement* annotator spent from 35 minutes to 1h20 annotating 15 sounds. Some participants put a lot of efforts exploring sibling categories in the hierarchy, making them waste time when considering the amount of refined labels (from 23 to 34 labels with a present validation). In contrast, the users of the AC Manual Annotator spent from 25 to 30 minutes performing the task.

Finally, it was observed that some participants gave a lot of importance to category sound examples and children, rather than relying on the name and textual description. This presents a risk since, in many occasions, neither the sound examples nor the listed children can be fully representative of a category diversity and complexity. It is therefore important that the tools promote the utilization of all the available information for annotators to take more solid and reliable decisions.

3.5 Conclusion

In this chapter we motivated the need for novel interfaces that facilitate the use of categories from large-scale taxonomies when annotating audio content. We presented the context of the Freesound Datasets initiative, which aims at creating openly available audio datasets. Two annotation interfaces were presented, which allow to target specific shortcomings when automatically generating labels. A preliminary evaluation with users allowed to evaluate our first versions of the tools and engage discussions.

Future work should focus on making the tasks faster, and aid the annotators on producing more exhaustive and consistent annotations. It will include improvements on the design, such as making the sound player more reachable to allow simultaneous exploration of category examples

and comparison with the audio resource being annotated. In addition, improved and more detailed task instructions should be designed, containing specific indications to make users focus on specific sound aspects covered by the taxonomy. These measures could help annotators to produce more comprehensive annotations.

Finally, the use of such tools by users when they upload content in to platforms such as Freesound has not been investigated. Even though the annotation processes proposed in this chapter seem more demanding than allowing users to use free form tags or textual descriptions, it seems that it could provide a more uniform way to annotate the content. Existing labels that were asked to be *refined* by the participants of the experiments were actually labels that were generated based on the tags already associated with the content. This refinement strategy could be proposed to the users of the platform, after having proposed some tags, which could produce more precise labels. This would benefit both users of the platform that would benefit from a more complete annotation of the content and the models that can be trained using predefined labels from an existing taxonomy.

Chapter 4

Audio Feature Performance Comparison for Unsupervised Sound Classification

Given the enormous amount of multimedia data available nowadays, retrieving and exploring content becomes difficult. Clustering is a possible solution for enabling data exploration. It can provide ways to find similarities within the data or discover underlying structures. Having clustering methods that can produce good performance for the diversity of content present in large sound databases is very challenging and relies substantially on the audio features used to represent the data. Recently, new kinds of audio representation have appeared through training deep neural networks with large amounts of data, resulting in better performance for the supervised classification of sounds. However, it is still unclear if these deep audio features can provide benefits within the unsupervised scenario, such as in sound clustering. In this chapter, we are interested in identifying which type of audio features is the most suited for clustering sounds present in large and diverse online sound collections. To this end, we compare the performance of five sets of audio features, using two different clustering algorithms. The evaluation is done using a collection of datasets, an outcome of previous chapters.

4.1 Introduction

The massive amount of content shared in online platforms challenges its successful exploration. Content-based methods, which rely on the content itself, are of great potential specially when accompanying metadata is incomplete or unstructured. Clustering is a fundamental technique of data mining that allows partitioning collections into groups. Clustering methods aim at unveiling hidden patterns by discovering natural groups in the data. It is often framed as an unsupervised classification problem, where no label associated to the content is known. Therefore it is considered one of the fundamental approaches in fields like bioinformatics or multimedia processing (Saxena et al., 2017). It differs from supervised classification which seeks to label objects with predefined classes.

In this chapter, we investigate the benefit of recent deep learning features for the unsupervised classification of sounds, specifically in the con-

text of real-world collaborative collections. We assess if supervised approaches relying on large annotated datasets are able to learn valuable re-usable audio features. Since annotated datasets are difficult and expensive to acquire, self-supervised approaches arise as an alternative with the potential to produce competitive features. Given the increasing amount of data generated and shared online, self-supervised approaches can take advantage of larger amounts of data compared to supervised approaches, which require manual curation. The next section provides an overview of related work. We present our experiment in Section 4.3 and the results in Section 4.4, followed by a conclusion section.

4.2 Related work

4.2.1 Audio features

Clustering can be understood as an unsupervised classification method. The first requirement for many supervised or unsupervised classification methods is to have a reliable feature representation, suitable for the particular data and application. For unsupervised methods, a similarity measure is often used. Generally, it is obtained by applying a distance metric on some numerical features. Numerous features have been developed, often tailored to suit specific applications and type of sounds.

The feature extraction step has often relied on feature engineering: the process of carefully designing features from low-level descriptors, relying on domain knowledge about the invariance of classes of sounds. Some are derived directly from the time domain audio representation. For instance, the *zero-crossing rate* allows the differentiation of periodic sounds, including musical instrument notes and noisy sounds like ocean waves (Peeters et al., 2011). Other features are derived from spectral representations of the sounds and are mostly motivated by the fact that human perception widely relies on the frequency content of sound signals. A great example showing the potential of feature engineering is the mel-frequency cepstral coefficients (MFCCs) feature, which were inspired by the human voice production mechanism and its auditory per-

ception. These coefficients allow a compact representation of the spectral envelope, that can efficiently represent a component of instruments' timbres (Herrera-Boyer et al., 2003), or more high-level features such as music mood (Kim et al., 2010). However, this type of spectral features do not convey temporal dynamics of the sound (Herrera-Boyer et al., 2003), which can be done by complementing them with temporal features in order to provide good performances for tasks such as instrument classification (Eronen & Klapuri, 2000). For music content, aspects related to the instrumentation, rhythmic structure and harmonic content can characterize specific music genres (Tzanetakis & Cook, 2002). Harmony can be represented using chroma features, which is robust to variation in timbre and thus allow to capture harmonic progressions independently from the instruments used (Muller & Ewert, 2010).

These sets of features allow representing audio in a high-dimensional space. However, having a large set of features increases the time and memory requirements of the learning algorithms and also degenerates the performances due to the curse of dimensionality and the existence of irrelevant, redundant and noisy dimensions. In order to mitigate this problem, feature selection is used for reducing the dimensionality by selecting a subset of the most relevant features. Feature selection has been shown to be an effective and efficient way to handle high-dimensional data (John et al., 1994). Moreover, it can improve performance, lower computational cost and improve model interpretability (Alelyani et al., 2013).

The temporal aspect of sounds is not often reflected in the audio features which are computed on relatively short frames of the audio signal. Representing audio clips often requires another abstraction, which generally consists in an aggregation step where the frame-based features are combined to produce a fixed-length numerical feature vector. Simple statistical representations which loose temporal relations and mix different components in a single representation are commonly used. Alternatives rely on using more advanced statistical models. Gaussian Mixture Models are able to have a better probabilistic representation (Li et al., 2003; Jensen et al., 2007), while extensions using Hidden Markov Model can capture temporal attributes (Herrera-Boyer et al., 2003). However in the

case of clustering where a distance measure has to be defined for comparing numerical audio representation vectors, using these probabilistic models is not always convenient, since defining efficient distance measures between probability densities is not straightforward and can be computationally expensive (Jensen et al., 2007).

Recently, techniques using Artificial Neural Networks have been able to provide an alternative to the handcrafted features previously developed. Tasks, such as auto-tagging or classification can be performed directly from the raw audio (Pons et al., 2017a), or using a spectrogram representation (Hershey et al., 2017). Furthermore, the internal representation that the neural network learns on one task, can be used for other applications, which is known as *transfer learning* (Choi et al., 2017). These types of approaches make use of pre-trained models as a starting point for different tasks. First layers from trained neural networks often learn similar features which can be applicable for many datasets and tasks (Yosinski et al., 2014). Intermediate layers can serve as a higher-level representations which can be used for instance in clustering (Jansen et al., 2017).

4.2.2 Clustering

Clustering is a type of unsupervised classification which consists in organising similar objects in groups called clusters. The clustering problem has been extensively addressed in the research community but is still a core challenge in the Information Retrieval field (Jain et al., 1999; Xu & Wunsch, 2005; Saxena et al., 2017). It is considered as an appealing approach for exploratory data analysis, as it can discover natural groupings or sets of patterns in data collections (Jain, 2010). Clustering approaches are based on optimising a metric derived from its objective definition: objects within a valid cluster should be more similar to each other than they are to an object belonging to a different cluster. The different approaches can be mainly divided into *partitional clustering* (such as K-means) and *hierarchical clustering* (such as agglomerative clustering) (Jain et al., 1999). In all the possible approaches, the content similarity measure involved is fundamental to the definition of a cluster.

When clustering sound collections, the features and distance measures are often chosen carefully for a given specific type of sounds, e.g. speech (Black & Taylor, 1997), musical instrument (Martins et al., 2007) and sound event recognition (Niessen et al., 2013)). This makes clustering sounds from online collections difficult, since the adopted features should be appropriate for the distinct types of content present in the collections. Moreover, the user-generated nature of online sound collections makes the content inconsistently distributed in terms of type and nature. As an example, in a collaborative audio database such as Freesound, we can find many instances of guitar sounds, while sitar sounds are rare. This can produce uneven densities in the feature space, which makes clustering approaches based on distance measures not always reliable (Roma et al., 2012). In addition, clustering algorithms are often fine-tuned by experts and researchers in order to provide good results on specific datasets, which cannot be achieved by users of online sound collections in the context of Search Result Clustering for instance. Furthermore, a large number of clustering algorithms are impractical due to their computational cost, specially when the clustering needs to be performed quickly to provide a good user experience.

Graph-based algorithms such as the ones relying on neighborhood graphs are a common method for dealing with large datasets (Fortunato, 2010; Liu et al., 2007). In particular, K-Nearest Neighbor Graphs can adapt to areas of different densities, since no fixed distance is assumed (Roma et al., 2012), which is suitable when dealing with unbalanced online collections. This type of graph can also help with the curse of dimensionality associated with the size of the feature space, as content features are not directly used in the clustering step (Marimont & Shapiro, 1979). In the case of images, large scale nearest neighbor searches enable scalability of such graph-based clustering methods (Liu et al., 2007). Graph partitioning, also referring to community detection, consists in dividing a graph into disjoint sets of nodes (Schaeffer, 2007). Among the different methods for automatically partitioning graphs (Fortunato, 2010), some approaches make use of a particular measure of the quality of a partition which is called modularity (Newman & Girvan, 2004). This measure is

proportional to the number of edges within clusters minus the expected number in an equivalent network with edges placed randomly. Although optimising this measure for finding a partition of the graph is a computationally hard task, some heuristic methods such as the Louvain method can provide decent performances in terms of quality and computational efficiency (Blondel et al., 2008). This makes these methods suitable for fast clustering purposes.

4.2.3 Clustering validation

In supervised classification, the evaluation of the resulting classification model is usually a straightforward process, and there are well-accepted evaluation measures and procedures, e.g., accuracy and cross-validation, respectively. However, clustering evaluation is not a well-developed or commonly used part of cluster analysis. Nonetheless, cluster evaluation is important and there exist different types of approaches which involve assessing the *appropriateness* of a partition after clustering, which is often called *cluster validation* (Arbelaitz et al., 2013; Liu et al., 2010). Clustering validation can be performed using *internal* or *external* criteria. On one hand, *internal* validation measures only rely on information from the data itself. They can be used to choose the best clustering algorithm as well as the optimal cluster number in an automatic way, without the use of any additional information (Halkidi & Vazirgiannis, 2001). These measures often rely on a mathematical formulation of the clustering objective related to what is referred as compactness or coherence (minimum intra-cluster variance) and separation or distinctiveness (inter-cluster density) (Liu et al., 2010). However, these evaluation methods rely on the same input features feeded to the clustering algorithms, and therefore do not allow comparing the performance of different audio representations. These measures are suitable for evaluating the clustering methods, once we already have a set of pre-defined features. On the other hand, when we have external information about the data, it is typically in the form of externally derived class labels for the data objects. In such cases, the usual procedure is to measure the degree of correspondence

between the cluster labels and the class labels. There exist some metrics that are based on classification performance measures such as Purity or F-measure (Manning et al., 2008). Other measures are based on the premise that any two objects that are in the same cluster should belong to the same class and two objects that are in two different cluster should belong to distinct classes. This type of measures, that can be referred as similarity measures between partitions, are the most widely used in the recent literature. They include Adjusted Rand Index (ARI) (Hubert & Arabie, 1985) and Adjusted Mutual Information (AMI) (Vinh et al., 2010). The literature suggests that AMI score is suited when the reference clustering (ground truth) is unbalanced and there exist small clusters. This corresponds often to what we have in collaborative audio collections where the content inconsistently distributed in terms of type and nature. On the contrary, when clusters are more balanced, ARI may be more appropriate (Romano et al., 2016).

Alternatively, evaluation methods based on human judgements may be carried out. These methods are appropriate for clustering tasks in which there is no objective solution, but they are more expensive and require careful design of user experiments. We carry out user-based evaluations of clustering algorithms in a real-world context in Chapter 7. In this section, we mainly focus on automatic evaluation, relying on external validation metrics by leveraging annotated data with content and tools presented in Chapter 2 and 3. As a complement, we perform a preliminary qualitative evaluation based on my personal judgment.

4.3 Experiment

4.3.1 Clustering methods

We propose to compare clustering performances using two different methods: K-means and a graph-based approach. In the graph-based approach, instead of directly using the features as input of a clustering method, we construct an intermediate representation of the data using a K-Nearest Neighbor Graph (KNN-Graph) (Dong et al., 2011). Each vertex repre-

sents a sound, and undirected edges connect each sound to its k most similar according to the euclidean distance in the feature space. Some preliminary empirical tests made us choose $\lfloor \log_2(N) \rfloor$ for the value of k , where N is the number of elements to cluster. This allows us to reach a sufficient number of neighbors for small collections, while limiting it for larger collections, which ensures low-computational complexity. Then, we use a community detection algorithm based on modularity optimisation for finding a partition of the graph (Blondel et al., 2008). For the K-means algorithm, in our experiments, we always set the number of clusters as equal to the number of classes in the dataset.

There are several reasons why we are interested in the graph-based approach. First, the number of clusters to obtain does not need to be specified unlike for the K-means algorithm. Then, it has been shown to be able to find clusters of different densities (Roma et al., 2012). Also, it can take advantage of nearest neighbors search techniques that can be fast to compute (e.g. (Cayton, 2008) or similar approximate methods (Aumüller et al., 2019)). Another advantage of these graph-based methods is their simplicity, which allows to use some interpretable heuristics for modifying the graph, its partition, or discarding clusters of low quality. The idea of discarding low quality clusters will be investigated in Chapter 7.

4.3.2 Audio features

In this work, we compare the performance of two clustering methods using five different sets of features. One set uses MFCC features, and the rest consist of pretrained deep learning embeddings taken from the literature. AudioSet embeddings use a spectrogram-based CNN architecture trained on a classification task on a large dataset containing millions of items and hundreds of semantic classes (Hershey et al., 2017; Gemmeke et al., 2017). OpenL3 also uses a spectrogram-based CNN architecture but is trained through self-supervised learning of audio-visual correspondence in videos (Cramer et al., 2019). Two models are available, OpenL3 music and OpenL3 env, which have been trained respectively with music and environmental content. SoundNet uses 1D convolutions directly on

the waveform, and was trained by transferring discriminative knowledge from a pretrained visual recognition network (Aytar et al., 2016). All these features are computed at a frame-based level and are then aggregated with arithmetic mean in order to obtain a fixed-sized feature vector for each clip.

4.3.3 Datasets

Since our main goal is to provide a comparison of clustering performances using different feature sets on diverse types of audio content, we leverage Freesound content for building multiple datasets that can represent the diversity in online sound collections. We make use of data gathered within Freesound Annotator to construct 44 datasets organized in 6 families comprising a total around 30k sounds. All sounds have a duration inferior to 10 seconds and most of them contain only one salient source, which, to some extent, mitigates the inconvenient of using a statistical aggregation over the frame-based features. Each family regroups datasets of similar theme as seen in Table 4.1. The classes are drawn from the AudioSet Ontology, a hierarchical taxonomy of sound-related concepts (Gemmeke et al., 2017). In our experiment, a dataset consists of one node in the taxonomy, and its labels are its direct children. This creates datasets that have different levels of specificity. For instance, inside the Nature family, one very broad dataset is the Nature dataset itself whereas Wind correspond to a more specific one. In total, our datasets present 215 different labels. Their distributions are kept unbalanced, in order to represent the non uniform distribution of content types present in online sound collections.

4.4 Results

4.4.1 Automatic evaluation

For evaluating the different set of features with the two clustering methods, we perform clustering on all the datasets and compute the validation

Dataset Families	Name of the datasets
Sound of Things	Bell, Alarm, Domestic, Door, Explosion, Engine, Glass, Tools, Mechanisms
Natural	Water, Natural, Liquid, Wind
Vehicle	Vehicle, Aircraft, Non-motorized land vehicle, Car, motor vehicle (road), Rail transport
Instruments	Plucked string, Bowed string, Mallet, Instruments, Wind, Percussion, Keyboard, Guitar, Musical concepts, Brass
Animals	Livestock, Domestic animals, Cat, Dog, Wild animals, Cattle & bovinæ
Human	Respiratory, Human voice, Singing, Human, Speech, Human group actions, Digestive, Hands, Human locomotion

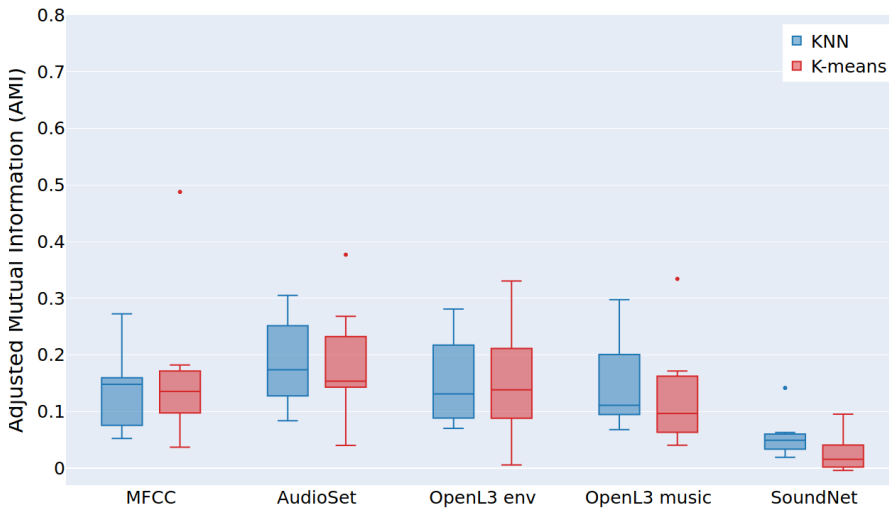
Table 4.1: Dataset families content.

score. We measure the similarity between the real partition (given by the ground truth labels) and the one given by the clustering methods by computing the Mutual Information score adjusted for chance (AMI) (Vinh et al., 2010). Table 7.3 displays the mean average scores for the different dataset families, audio features and methods. Figure 4.4.1 represent box-plots of the AMI score on the different dataset families and on all the datasets. Random partitions have an AMI close to 0 and perfect labelings have a score of 1. This allows us to compare the performance of the different features on different types of audio samples and at different levels of specificity. A discussion about these results is given in Sec. 4.4.3 The code for replicating the clustering is available at: <https://github.com/xavierfav/feature-comparison-clustering>

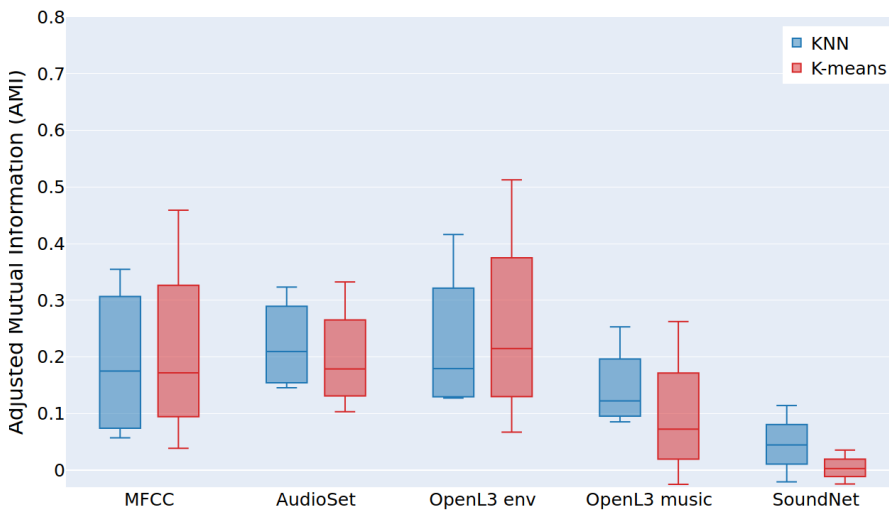
Dataset	MFCC		AudioSet		OpenL3 music		OpenL3 env		SoundNet	
	K-means	KNN	K-means	KNN	K-means	KNN	K-means	KNN	K-means	KNN
Sound of things	0.159	0.131	0.186	0.191	0.128	0.148	0.148	0.154	0.026	0.054
Natural Sounds	0.210	0.190	0.198	0.222	0.096	0.146	0.252	0.225	0.004	0.046
Vehicles	0.176	0.097	0.308	0.204	0.128	0.136	0.082	0.143	-0.002	0.030
Instruments	0.155	0.147	0.180	0.191	0.171	0.216	0.150	0.208	0.021	0.072
Animals	0.281	0.156	0.162	0.199	0.174	0.160	0.195	0.166	0.003	0.075
Human Sounds	0.119	0.124	0.225	0.212	0.137	0.159	0.163	0.169	0.019	0.067
Average	0.184	0.141	0.210	0.203	0.139	0.161	0.165	0.178	0.012	0.057

Table 4.2: Average performance (AMI) across the different dataset families of the K-means and the KNN-Graph clustering methods with the different features. An AMI close to 0 corresponds to a random partition while perfect matches gives 1 AMI.

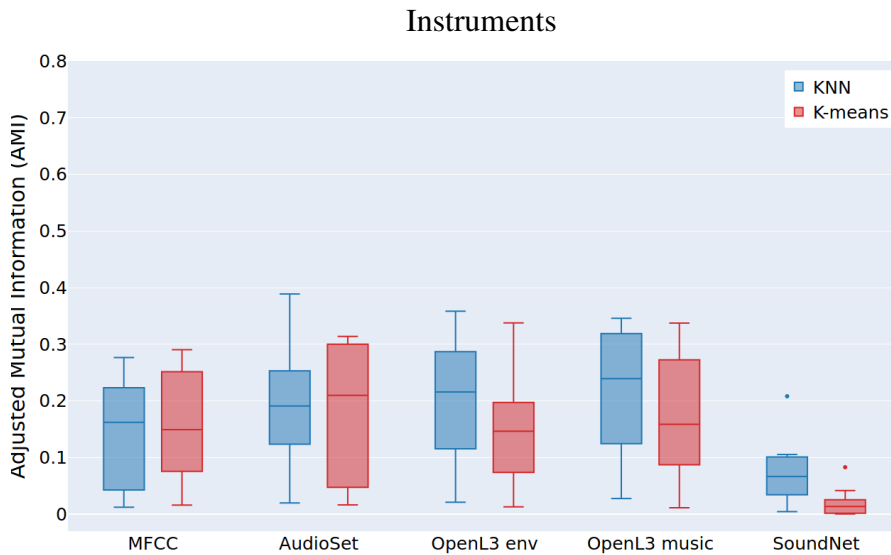
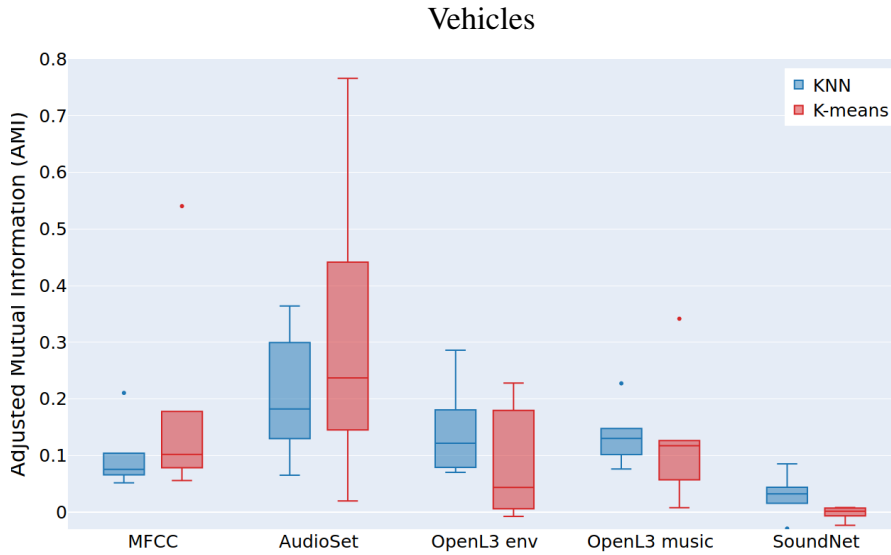
Sound of things



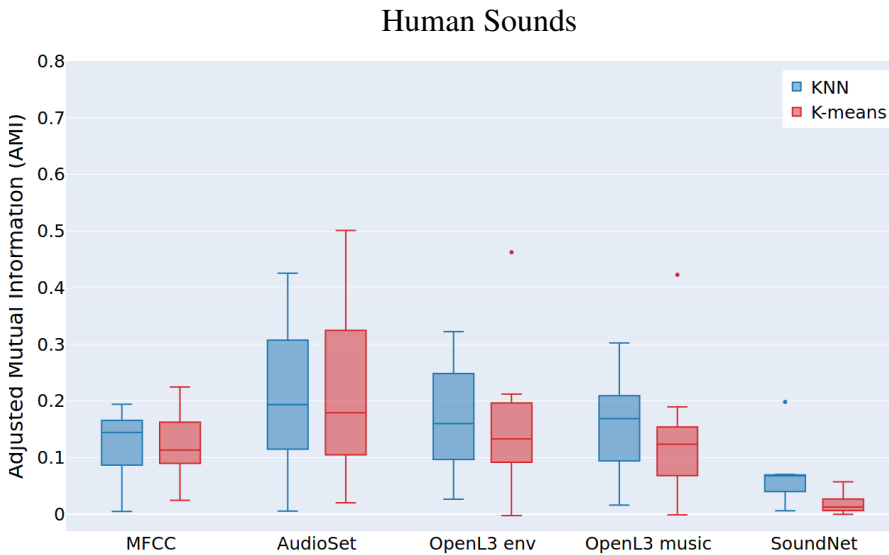
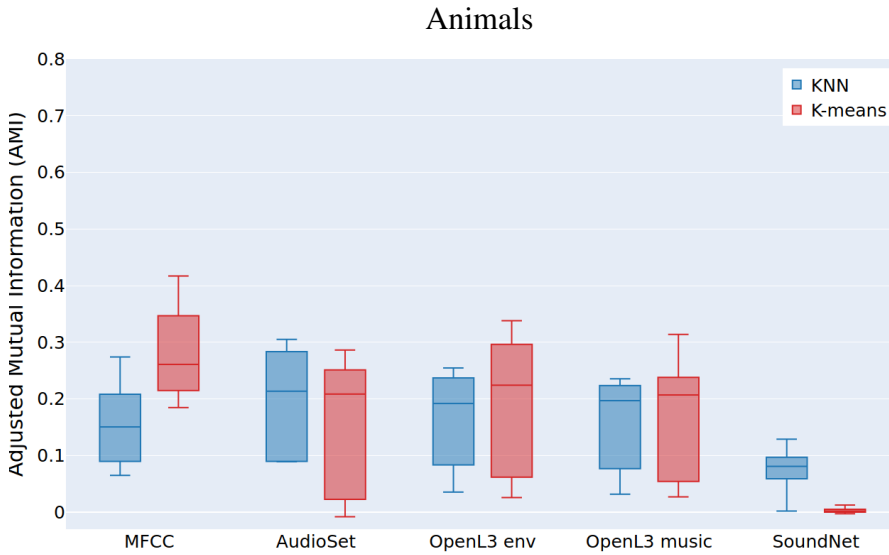
Natural Sounds



Chapter 4. Audio Feature Performance Comparison for Unsupervised Sound Classification



Chapter 4. Audio Feature Performance Comparison for Unsupervised Sound Classification



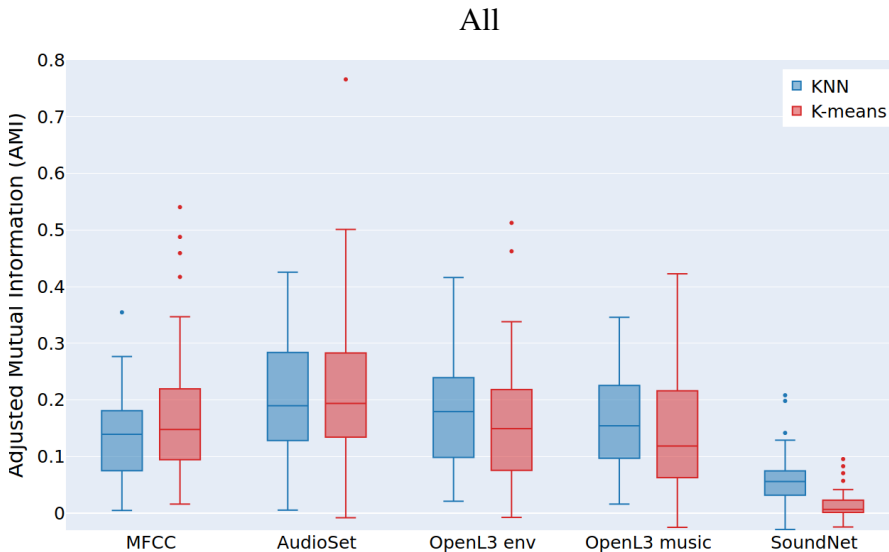


Figure 4.1: Box-plots of the Adjuster Mutual Information scores re-grouped by dataset family and for all the datasets.

4.4.2 Qualitative evaluation

In order to complement the automatic evaluation, we performed some qualitative evaluation of the performance of the different features with the graph-based clustering method (KNN) only. The reasons why we do not perform such analysis with the K-means algorithm are because of time constrains and based on the intuition that the KNN approach is more efficient and computationally efficient. We qualitatively evaluate our approach by manually inspecting the clustering results performed on the different datasets. For this purpose, we use an interface that displays nodes and edges in a two-dimensional space using a force-directed algorithm for computing the layout¹ (Eades, 1984; Eades & Klein, 2018),

¹Also called as Spring layout. Nodes are modeled as physical objects that mutually exert forces on each other. We use the d3 javascript library's implementation: <https://github.com/d3/d3-force>.

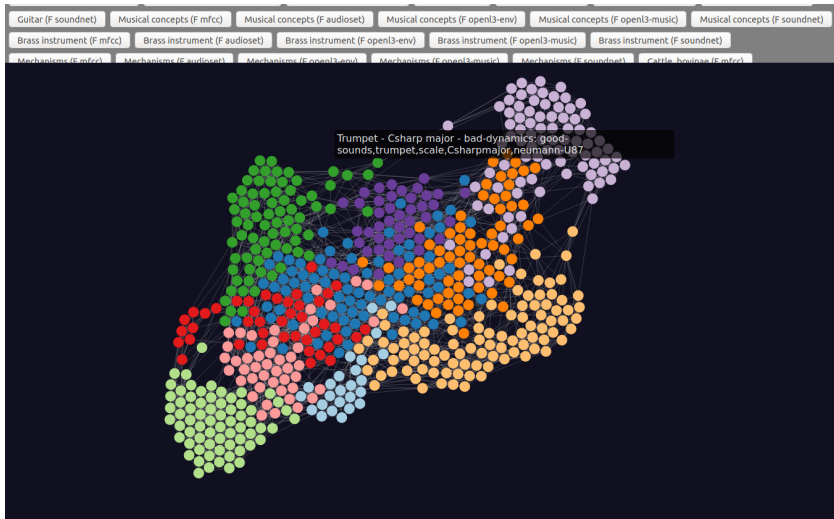


Figure 4.2: Visualisation of the clustered graph for the "Brass instrument" dataset using the AudioSet features. The graphs for different datasets and features can be explored from a browser at this url: <https://xavierfav.github.io/feature-comparison-clustering/web-visu/>

and allows to listen to the sounds (Figure 4.2). The clustered graphs for the different datasets are made available from the browser at this address: <https://xavierfav.github.io/feature-comparison-clustering/web-visu/>. In this page, buttons at the top allow you to choose the dataset and the feature set used for the graph creation and its partition. When hovering the nodes with the mouse, sounds are played, and the name of the audio clip and its tags is displayed. The colors correspond to the assigned clusters. You can zoom in and out using the mouse wheel, and move in the space by dragging.

In order to get some insights about how the different feature sets perform in combination with the clustering algorithms, we explored the graphs created for different datasets by visualizing them and listening to the sounds. Qualitative observations for three datasets (Guitar, Wind instruments, Natural sounds and Domestic sounds, home sounds) are listed

bellow:

1. **Guitar** corresponds to a dataset containing sounds that are relatively specific and not very varied.
 - (a) **MFCC**
 - A lot of clusters appear very separated.
 - Very similar sounds are often clustered together, e.g. string harmonics, metal distorted power chords, strums. Many of these clusters consist of sounds of a single instrument recorded several times.
 - The system seems able to cluster sounds that have different spectral balances, e.g. bass guitar sounds are well separated from acoustic guitars.
 - However we can often see very different content mixed in a cluster.
 - (b) **AudioSet**
 - A lot of clusters look much more closer to each other.
 - The content of the clusters seems a bit more diverse and include sounds from different instruments and users. The results are quite coherent, e.g. distorted sounds, acoustic guitars, clean electric guitars appear in distinct clusters.
 - (c) **OpenL3 env**
 - The clusters appear much more separated. Similarly to the MFCC features, very similar sounds are clustered together.
 - However, the system is able to cluster most of the bass guitar sounds together, even if they don't come from the same user and instrument.
 - Clusters appear a bit more fuzzy and overlapped in the middle of the graph.
 - (d) **OpenL3 music**
 - Similarly to the OpenL3 env features, we get many clusters that are quite separated from the others and that correspond to very similar sounds often produced by one user.

- (e) **SoundNet**
 - Very similar sounds are clustered together.
 - However, several types of sounds get mixed together in large clusters, e.g. bass, guitar strums, single note, slides, ...
- 2. **Wind instruments** corresponds also to a dataset containing sounds that are relatively specific with a low range of different types.
 - (a) **MFCC**
 - Very similar sounds are clustered together but mixed with different instruments, e.g. flute with oboe.
 - Different flutes are not in the same cluster.
 - (b) **AudioSet**
 - Again, many clusters appear closer to each other.
 - Clarinet sounds are mixed with saxophone sounds.
 - A bit fuzzy in the middle with many things mixed together, presence of a considerable amount of weird timbre, e.g. clarinet sounding like high pitch oboe note.
 - Single notes are clustered together, full scales are also together.
 - (c) **OpenL3 env**
 - Again clusters appear quite separated.
 - Specific playing techniques appear in different clusters, e.g. staccato, legato, single notes, scales.
 - (d) **OpenL3 music**
 - Clusters appear even more separated.
 - Again it separates specific playing techniques.
 - (e) **SoundNet**
 - Clusters are close to each other and do not appear very separated.
 - They actually contain very different content.

3. **Natural sounds** consists in a dataset that contains a broader range of types of sounds.
 - (a) **MFCC**
 - Sounds with similar timbre or color are sometimes in a same clusters.
 - However, semantically different sounds are often mixed in the same cluster.
 - Clusters appear to be quite separated in the graph.
 - (b) **AudioSet**
 - Clusters look a bit more separated.
 - More semantically coherent.
 - The system is able to also cluster sounds according to some perceptual characteristics, e.g. strong and light water flows or stream.
 - Sometimes the system clusters together semantically different sounds but that convey similar perceptual features, e.g. fire crackles and water splash.
 - (c) **OpenL3 env & music**
 - Clusters look much closer to each other, which was not the case for musical sounds from previous datasets.
 - As a result, some clusters contains different types of sound.
 - (d) **SoundNet**
 - Clusters look relatively close to each other.
 - Clusters are mixed with semantically different sounds and do not seems to regroup sounds with similar perceptual features.
4. **Domestic sounds, home sounds** consists in a dataset that contains a broad range of types of sounds.
 - (a) **MFCC**
 - Sounds with similar timbre or color are sometimes in a same

clusters. For instance rustling sounds such as coin dropping, teaspoon clinks, keys jangling, scissors or knife sharpening are grouped in one cluster.

- Relatively short sounds are often separated from sustained or repetitive ones.
- Clusters contain semantically different types of sound.

(b) **AudioSet**

- The partition is sometimes more consistent semantically, e.g. one cluster contains a lot of water sounds.
- A lot of clusters seem to overlap between each others.

(c) **OpenL3 env & music**

- There still exist some overlap between clusters but they look more separated.

(d) **SoundNet**

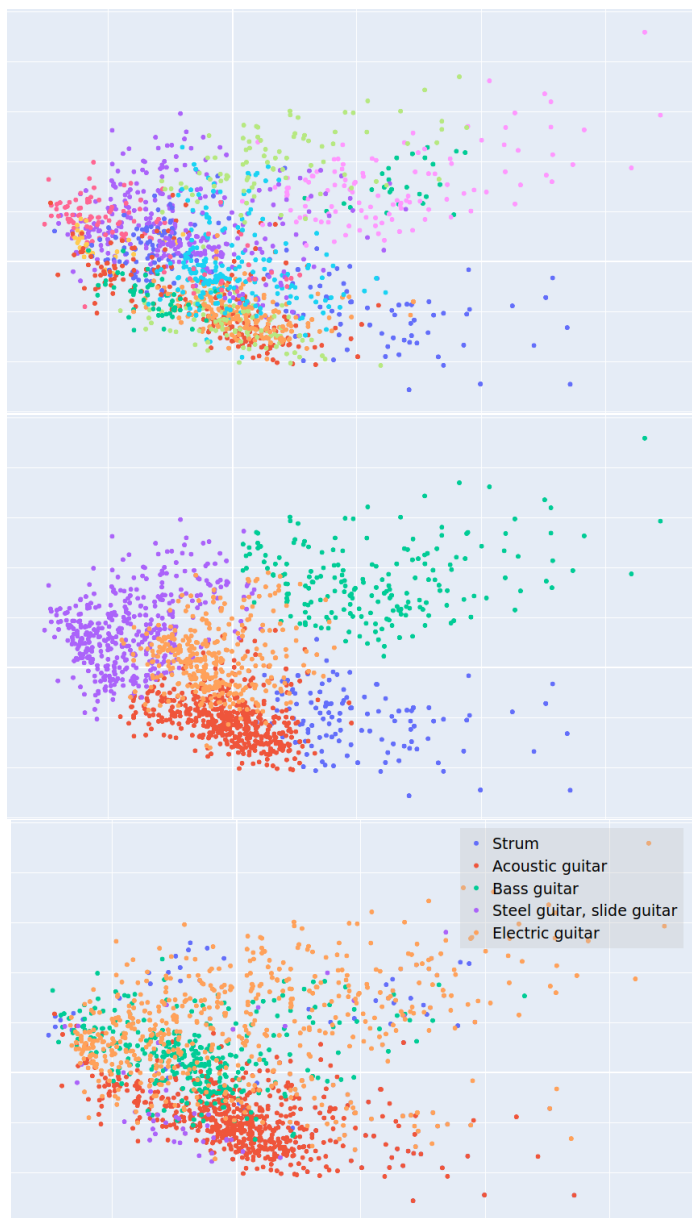
- The structure of the graph does not seem to correspond to any perceptual or semantic clue.

4.4.3 Discussion

Performance

Overall, we observe that the AudioSet features produce the best mean average AMI. MFCC and OpenL3 features can in some cases provide the best performance, whereas SoundNet leads to bad performance with results close to random partitions. AudioSet provides clear benefits in the Human Sounds, Vehicles and Sound of Things families compared to the other features. The corresponding datasets can include a wide range of complex sounds where the distinction between some classes can be hard to achieve even for human listeners. Recognizing cars among truck or bus sounds can be sometimes difficult. Within the Sound of Things family, a wide class variability can make the approach to produce clusters different from the ground truth labels. For instance, the Tools dataset can include

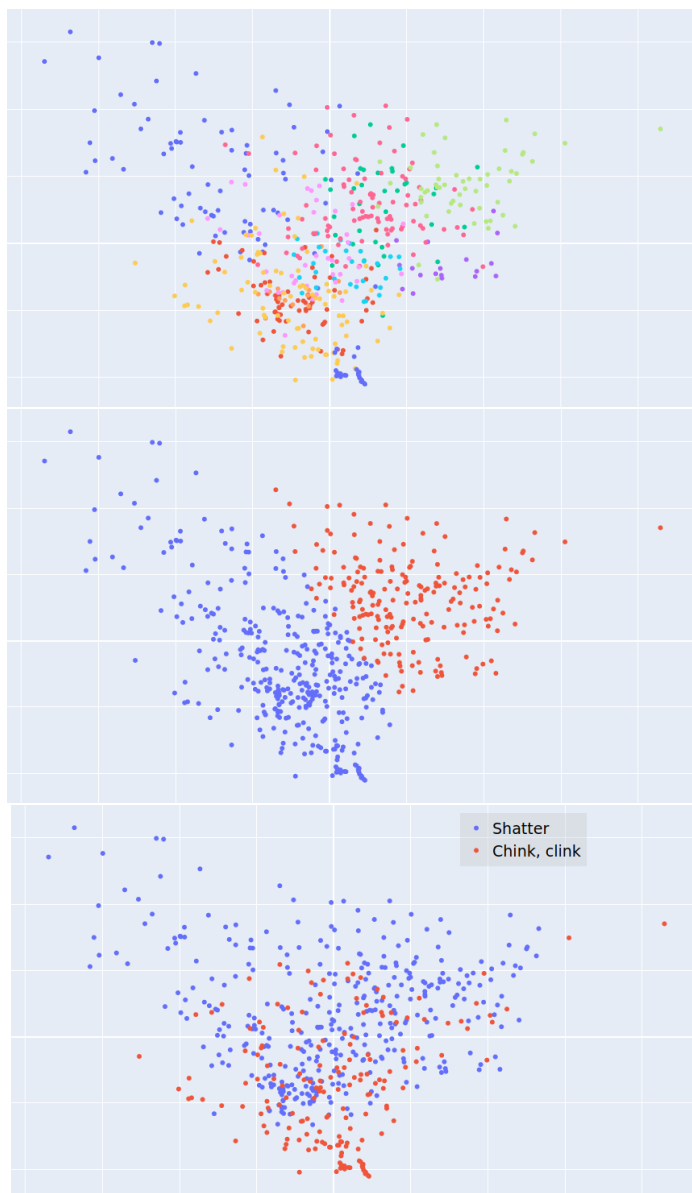
Guitar



Wind instrument, woodwind instrument



Glass



Natural Sounds



Human Sounds

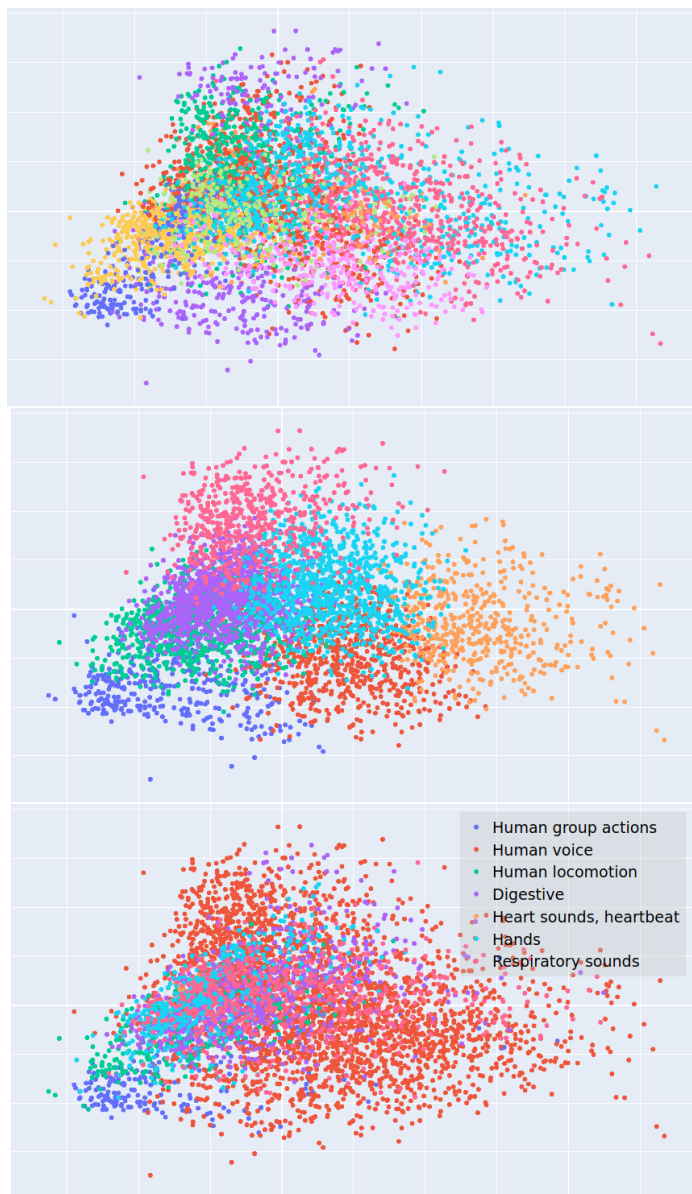


Figure 4.3: First two components of the PCA decomposition of the audioset embeddings for the different datasets. The first two plots' colors represent clusters obtained with the KNN and the K-means approaches respectively, the last one displays the ground truth labels.

electric or mechanical sawing machines, hammers can be used to blow metallic or wooden object that would produce very particular sounds. In these scenarios, AudioSet features that were learned on a supervised classification task seem more suited in order to provide semantically coherent clusters. These features are trained on a supervised classification task on a large dataset labeled with hundreds of semantic categories. They enable to obtain clusters for complex sounds that seem to be hard to obtain when using MFCC features or embedding trained in a self-supervised manner. However, it is worth mentioning that AudioSet features were learned on a classification task using a dataset which uses labels that also exist in the evaluation datasets we use here, but with different sounds. Therefore, they are logically well suited for producing clusters that are similar to the ground truth labels in our datasets.

OpenL3 music features show better performance for the Instruments datasets family, which suggests that training a model within the specific context of sounds (in this case musical instruments) can lead to better performances on related data. Similarly, when using the OpenL3 features trained on the environmental dataset (OpenL3 env), we observe better clustering performance on the Natural datasets family. This suggests that having specific models adapted for different contexts can lead to better performance compared to using only one general model. Training an embedding model with content from a specific range of sounds can allow to get more specialized models. However, for all the other families, we don't observe major differences in performances for the two OpenL3 models.

As a complement to the external evaluation, a qualitative evaluation based on my personal judgment is provided. In some cases, MFCC features are able to produce sometimes coherent clusters according to some perceptual low-level features related to timbre and color of the sounds. Moreover, this can sometimes provide good performance in term of semantic coherence. This is the case for instance when the dataset contain classes that have very distinct low-level characteristics, such as string harmonics, distorted power chords and strums within some guitar sounds. However, it can in other cases produce cluster that do not keep semantic coherence for instance for the Domestic sounds, home sounds dataset.

On the contrary, it seems that deep learning features are able to produce much more coherent clusters for a larger number of datasets. The automatic evaluation, to some extent, corresponds to the qualitative observations. For instance in the case of natural sounds, MFCC features seem to produce better results than in the case of domestic sounds, home sounds, where many different types of sounds are mixed in one cluster. Finally, MFCC features seem adequate for Animal sounds, for which such a spectral representation seems sufficient for describing animal vocalizations.

Similarity notion

When using MFCC features, the system seems to be able to group together very similar sounds, such as some produced by one single instrument recorded in similar conditions. This happens particularly a lot with instrument sounds, where users from Freesound upload various recordings of a single instrument in a specific recording condition, often organized in packs ². This creates very coherent clusters that are too specific and fail to cover a wider range of samples from a same or similar instruments. This effect of having clusters that contain very particular instances of a class seems mitigated when using the AudioSet or the OpenL3 features. These features often produce larger or less separated clusters and can cover a wider range of variability within a class. For instrument sounds, our qualitative inspection of the clusters suggests that the AudioSet features are the ones that produce better clusters in terms of covering a semantic class variability. The fact that AudioSet consist of embeddings training on a supervised classification task in order to recognize sound events such as instrument notes seem to make them more appropriate for providing a wider notion of similarity that seem to be more appropriate for providing informative clusters for exploring sounds.

²A lot of the largest packs consist of musical instrument samples https://freesound.org/browse/packs/?order=-num_sounds

Graph structure

Interestingly, it happens that when using the MFCC or the SoundNet features, we obtain graphs that have a visual structure with very separated clusters. However, when inspecting the sounds, these clusters often mix different types of content. On the contrary, when using the AudioSet features, when clusters look visually separated, it often seems that the clusters are more coherent. And, when some clusters look fuzzy and seem to visually overlap, they are often not so coherent and they contain a wider range of sounds. This means that when partitioning the graph built using the MFCC or the SoundNet features, the algorithm was able to optimize the clustering objective (*modularity*) well, meaning that the partition obtained according to the considered features seems of good quality. However, the clustering performance is not as good according to the external and qualitative evaluations. The visually well-defined clusters are often not coherent and contain very dissimilar sounds. This suggests that the MFCC and SoundNet features contain noise that hinders the performance of the clustering system. However, the graph structure seems to correspond to the quality of the clustering when using the AudioSet features. This can be promising for instance for automatically assessing its quality from the *modularity* or the graph structure.

Clustering method

According to the quantitative evaluation, the K-means algorithm is producing slightly better results than the KNN approach. However, from Figure 4.1, the performance score is less consistent and tends to vary more across the different datasets. Moreover, the K-means algorithm needs to be specified with the number of clusters in advance, which was automatically set to the number of classes in each datasets for this experiments. When we look at the visualisation Figure 4.3, we observe that the KNN-based approach tend to produce more clusters than the existing classes in the dataset. And when listening to the sounds from different clusters in the graph, we sometimes find clusters that are more precise than the ground truth labels. Additionally, K-means seem to fail to discover classes that

appear to be overlapping when looking at the first two PCA components. This is clear for instance in the Wind instrument, woodwind instrument and the Guitar datasets, where the different instruments are overlapping in the 2D space. As a reminder, K-means is computed directly on the feature sets, and not after the PCA transformation. On the contrary, the KNN-based approach seem to be able to discover clusters that appear as overlapped in the PCA visualisations.

4.5 Conclusion

With the advancement of machine learning, novel features appeared that improve the performance of supervised classification methods. In this work we evaluated some of these novel features in a clustering scenario using many datasets which reflect a large variety of types of sounds present in online collections.

We demonstrate that novel deep learning features can be used for clustering diverse sound collections and achieve competitive or superior performance compared to more traditional features such as MFCC. Moreover, this study suggests that spectrogram-based convolutional architectures trained on a supervised task using a large taxonomy of semantic concepts can provide features with better performance for clustering complex sounds. Intermediate layers of the networks convey high-level semantics which makes them suitable for obtaining a similarity metric adapted for a large variety of sounds. Additionally, training features in a specific domain can also provide better performance within this restricted domain. Finally, an advantage of deep learning features seems to be that they can compute accurate features on larger audio frames. As an example, AudioSet embeddings consist of a 128-sized feature vector calculated over a window of 1 second, whereas MFCC features are typically computed on tens or hundreds millisecond frames.

Learning features with large annotated dataset seems provide audio representations that can produce better clustering. However, building these datasets is difficult and requires a considerable amount of effort.

In order to leverage larger amount of data, unsupervised approaches seem promising, but do not yet reach the same performance.

Handcrafted features such as MFCCs are in some case able to produce good clustering, but they sometimes fail to be able to capture semantic properties conveyed by audio signals.

The graph-based clustering methods has several advantages over the traditional K-means algorithm. First, it does not require to specify the number of clusters in advance. Moreover, unlike K-means, it is able to discover clusters that are more consistent with the ground truth labels. In particular, it is able to identify clusters that are not linearly separable according to the two dimensions with maximum variability. Approaches based on distances to centroids attribute a higher importance to dimensions with high variability, which may not correlate with the separation between categories.

Chapter 5

Co-Aligned Autoencoders for Learning Semantically Enriched Audio Representations

Audio representation learning based on deep neural networks (DNNs) emerged as an alternative approach to handcrafted features. In the last chapter we saw that DNNs can produce features that provide good performance for the unsupervised classification of sounds. However, for achieving such performance, these DNNs need to be trained with a large amount of annotated data which can be difficult and costly to obtain. In this chapter, we introduce COALA, which stands for Co-Aligned Autoencoders for Learning Semantically Enriched Audio Representations. This method learns audio representations, by aligning the learned latent representations of audio and associated tags taken from an online sound sharing platform. The alignment is done by maximizing the agreement of the latent representations of audio and tags, using a *contrastive* loss function. The result is an audio embedding model which reflects acoustic and semantic characteristics of sounds. We evaluate the quality of our embedding model, measuring its performance as a feature extractor on three different tasks (namely, sound event recognition, music genre classification and musical instrument classification), and investigate what type of characteristics the model captures. Our results are promising, sometimes in par with the state-of-the-art in the considered tasks. Furthermore, the embeddings produced with our method are well correlated with some acoustic descriptors.

5.1 Introduction

Traditional audio-based machine learning models were trained using sets of handcrafted features. Recent approaches based on deep learning (DL) are able to learn such features directly from the data. Achieving high performance with DL-based methods and models, often requires sufficient labeled data which can be difficult and costly to obtain, especially for audio signals (Favory et al., 2018). As a way to lift the restrictions imposed by the limited amount of audio data, different published works employ transfer learning on tasks where only small datasets are available (Yosinski et al., 2014; Choi et al., 2017). Usually in such a scenario, an embedding

model is first optimized on a supervised task for which a large amount of data is available. Then, this embedding model is used as a pre-trained feature extractor, to extract input features that are used to optimize another model on a different task, where a limited amount of data is available (Van Den Oord et al., 2014; Choi et al., 2017; Pons & Serra, 2019a; Alonso-Jiménez et al., 2020).

Recent approaches adopt self-supervised learning, aiming to learn audio representations on a large set of unlabeled multimedia data, e.g. by exploiting audio and visual correspondences (Aytar et al., 2016; Arandjelovic & Zisserman, 2017). Such approaches have the advantage of not requiring manual labelling of data, and have been successful for learning audio features that can be used in training easy-to-use, but competitive classifiers (Cramer et al., 2019). Other approaches focus on learning audio representations by employing a distance metric learning strategy and weakly annotated data. For example, by utilizing the triplet-loss to maximize the agreement between different songs of a same artist (Park et al., 2017) or by using or another contrastive loss for maximizing the similarity of different transformations of the same example (Chen et al., 2020). Other approaches leverage images and their associated tags to learn content-based representations by aligning autoencoders (Schonfeld et al., 2019).

In our work we are interested in learning audio representations that can be used for developing general machine listening systems suited for a wide range of types of sounds, instead of focusing on one sort of sounds in particular. We take advantage of the massive amount of online audio recordings and their accompanying tag metadata, and learn acoustically and semantically meaningful features. To do so, we propose a new approach inspired from the image and the natural language processing fields (Schonfeld et al., 2019; Silberer & Lapata, 2014), but we relax the alignment objective by employing a contrastive loss (Chen et al., 2020). This allows a co-regularization of the latent representations of two autoencoders, each one learned on a different modality. The contributions of our work are:

- We adapt a recently introduced contrastive learning framework (Chen

et al., 2020), and we apply it for audio representation learning in an heterogeneous setting (the embedding models process different modalities).

- We propose a learning algorithm, combining a contrastive loss and an autoencoder architecture, for obtaining aligned audio and tag latent representations that reflect both semantic and acoustic characteristics.
- We provide a thorough investigation of the performance of the approach, by employing three different classification tasks.
- Finally we conduct a correlation analysis of our embeddings with acoustic features in order to get more understanding of what characteristics they capture.

The rest of the chapter is organized as follows. In Section 5.2 we present our proposed method. Section 5.3 describes the utilized dataset, the tasks and metrics that we employed for the assessment of the performance, the baselines that we compare our method with, and the correlation analysis with acoustic features that we conducted. The results of these evaluation processes are presented and discussed in Section 5.4. Finally, Section 5.5 concludes the chapter and proposes future research directions.

5.2 Co-aligned autoencoders

Our method employs two different autoencoders (AEs) and a dataset of multi-labeled annotated (i.e. multiple labels/tags per example) time-frequency (TF) representations of audio signals, $\mathbb{G} = \{(\mathbf{X}_a^q, \mathbf{y}_t^q)\}_{q=1}^Q$, where $\mathbf{X}_a^q \in \mathbb{R}^{N \times F}$ is the TF representation of audio, consisting of N feature vectors with F log mel-band energies, $\mathbf{y}_t^q \in \{0, 1\}^C$ is the multi-hot encoding of tags for \mathbf{X}_a^q , out of a total of C different tags, and Q is the amount of paired examples in our dataset. These tags characterize the content of each corresponding audio signal (e.g. “kick”, “techno”, “hard”).

The audio TF representation and the associated, multi-hot encoded

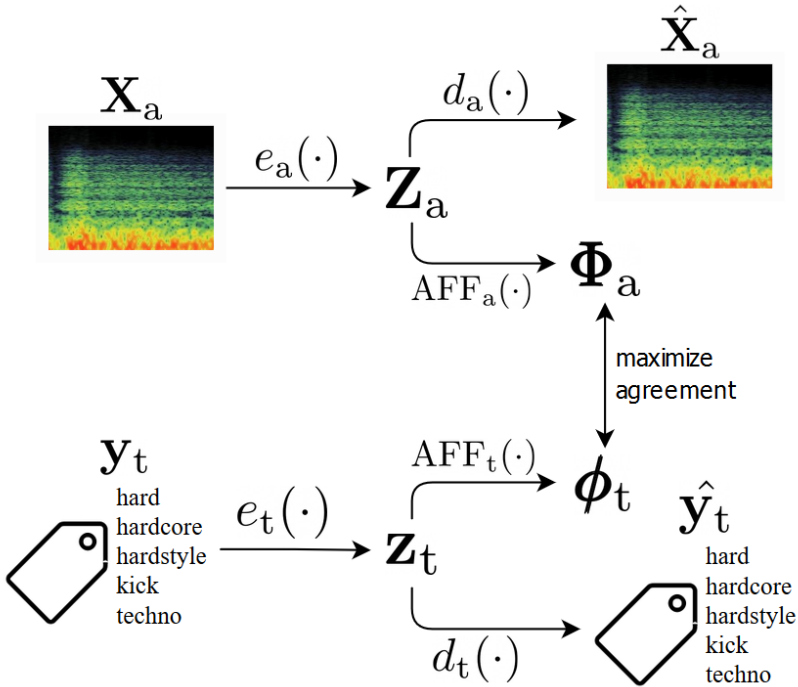


Figure 5.1: Illustration of our proposed method. Z_a and z_t are aligned through maximizing their agreement and, at the same time, are used for reconstructing back the original inputs.

tags of the audio signal, are used as inputs to the two different AEs, one targeting to learn low-level acoustic features for audio and the other learning semantic features (for the tags), by employing a bottleneck layer and a reconstruction objective. At the same time, the learned low-level features of the audio signal are aligned with the learned semantic features of the tags, using a contrastive loss. All employed modules are jointly optimized, yielding an audio encoder that provides audio embeddings which capture both low-level acoustic characteristics and semantic information regarding the contents of the audio. An illustration of our method is in Figure 5.1.

5.2.1 Learning low-level audio and semantic features

For learning low-level acoustic features from the input audio TF representation, \mathbf{X}_a ¹, we employ a typical AE structure based on convolutional neural networks (CNNs) and on having a reconstruction objective. Since AEs have proven to be effective in unsupervised learning of low-level features in different tasks and especially in audio (Van Den Oord et al., 2017; Amiriparian et al., 2017; Mimilakis et al., 2018; Drossos et al., 2018), our choice of the AE structure followed naturally.

The AE that processes \mathbf{X}_a is composed of an encoder $e_a(\cdot)$ and a decoder $d_a(\cdot)$, parameterized by θ_{ea} and θ_{da} , respectively. e_a accepts \mathbf{X}_a as an input and yields the learned latent audio representation, $\mathbf{Z} \in \mathbb{R}_{\geq 0}^{T' \times F'}$. Then, d_a gets as an input \mathbf{Z}_a and outputs a reconstructed version of \mathbf{X}_a , $\hat{\mathbf{X}}_a$, as

$$\mathbf{Z}_a = e_a(\mathbf{X}_a; \theta_{ea}), \text{ and} \quad (5.1)$$

$$\hat{\mathbf{X}}_a = d_a(\mathbf{Z}_a; \theta_{da}). \quad (5.2)$$

We model e_a using a series of convolutional blocks, where each convolutional block consists of a CNN, a normalization process, and a non-linearity. As a normalization process we employ the batch normalization (BN), and as a non-linearity we employ the rectified linear unit (ReLU). The process for each convolutional block is

$$\mathbf{H}^{l_{ea}} = \text{ReLU}(\text{BN}^{l_e}(\text{CNN}^{l_e}(\mathbf{H}^{l_e-1}))), \quad (5.3)$$

where $l_{ea} = 1, \dots, N_{\text{CNN}}$ is the index of the convolutional block, $\mathbf{H}^{l_{ea}} \in \mathbb{R}_{\geq 0}^{K_{l_{ea}} \times T'_{l_{ea}} \times F'_{l_{ea}}}$ is the $K_{l_{ea}}$ learned feature maps of the l_{ea} -th CNN, $\mathbf{H}^{N_{\text{CNN}}} = \mathbf{Z}_a$, and $\mathbf{H}^0 = \mathbf{X}$.

Audio decoder, d_a , is also based on CNNs, but it employs transposed convolutions (Radford et al., 2016; Dumoulin & Visin, 2016) in order to expand \mathbf{Z} back to the dimensions of \mathbf{X} . For having a decoding scheme

¹For the clarity of notation, the superscript q is dropped here and for the rest of the document, unless it is explicitly needed.

analogous to the encoding one, we employ another set of N_{CNN} convolutional blocks for d_a , again with BN and ReLU, and using the same serial processing described by Eq. (5.3). This processing yields the learned feature maps of the decoder, $\mathbf{H}^{l_{da}} \in \mathbb{R}_{\geq 0}^{K_{l_{da}} \times T'_{l_{da}} \times F'_{l_{da}}}$, with $l_{da} = 1 + N_{\text{CNN}}, \dots, 2N_{\text{CNN}}$ and $\mathbf{H}^{2N_{\text{CNN}}} = \hat{\mathbf{X}}_a$. To optimize e_a and d_a , we employ the generalized KL divergence, D_{KL} , and we utilize the following loss function

$$\mathcal{L}_a(\mathbf{X}_a, \theta_{ea}, \theta_{da}) = D_{\text{KL}}(\mathbf{X}_a || \hat{\mathbf{X}}_a). \quad (5.4)$$

Each audio signal represented by \mathbf{X}_a is annotated by a set of tags from a total of C possible tags. We want to exploit the semantics of each tag and, at the same time, capture the semantic relationships between tags. For that reason, we opt to use another AE structure, which outputs a latent learned representation of the set of tags of \mathbf{X}_a as the learned features from the tags, and then tries to reconstruct the tags from that latent representation. Similar approaches have been used in (Silberer & Lapata, 2014), where an AE structure was employed in order to learn an embedding from a k -hot encoding of tags/words that would encapsulate semantic information. Specifically, we represent the set of tags for \mathbf{X} as a multi-hot vector, $\mathbf{y}_t \in \{0, 1\}^C$. We use again an encoder e_t and a decoder d_t , to obtain a learned latent representation of \mathbf{y}_t as

$$\mathbf{z}_t = e_t(\mathbf{y}_t; \theta_{et}), \text{ and} \quad (5.5)$$

$$\hat{\mathbf{y}}_t = d_t(\mathbf{z}_t; \theta_{dt}), \quad (5.6)$$

where $\mathbf{z}_t \in \mathbb{R}_{\geq 0}^M$ is the learned latent representation of the tags for \mathbf{X} , \mathbf{y}_t and $\hat{\mathbf{y}}_t$ is the reconstructed multi-hot encoding of the same tags \mathbf{y}_t . The e_t consists of a set of trainable feed-forward linear layers, where each layer is followed by a BN and a ReLU, similar to Eq. 5.3. That is, if FNN^{l_t} is the l_t -th feed-forward linear layer, then

$$\mathbf{h}_t^{l_t} = \text{ReLU}(\text{BN}^{l_t}(\text{FNN}^{l_t}(\mathbf{h}_t^{l_t-1}))), \quad (5.7)$$

where $l_t = 1, \dots, N_{\text{FNN}}$, $\mathbf{h}_t^{N_{\text{FNN}}} = \mathbf{z}_t$, and $\mathbf{h}_t^0 = \mathbf{y}_t$. To obtain the reconstructed version of \mathbf{y}_t , $\hat{\mathbf{y}}_t$, through \mathbf{z}_t , we use the decoder d_t , which

is modeled analogously to e_t and containing another set of N_{FNN} feed-forward linear layers. d_t processes \mathbf{z}_t similarly to Eq. 5.7, with $\mathbf{h}_t^{1+N_{\text{FNN}}}$ to be the output of the first feed-forward linear layer of d_t , and $\mathbf{h}_t^{2N_{\text{FNN}}} = \hat{\mathbf{y}}_t$. To optimize e_t and d_t we utilize the loss $\mathcal{L}_t(\mathbf{y}_t, \theta_{e_t}, \theta_{d_t}) = CE(\mathbf{y}_t, \hat{\mathbf{y}}_t)$, where CE is the cross-entropy function.

5.2.2 Alignment of acoustic and semantic features

One of the main goals of our method is to infuse semantic information from the latent representation of tags to the learned acoustic features of audio. To do this, we maximize the agreement between (i.e align) the paired latent representations of the audio signal, \mathbf{Z}_a^q , and the corresponding tags, \mathbf{z}_t^q , by using a contrastive loss. Aligning these two latent representations (by pushing \mathbf{Z}_a^q towards \mathbf{z}_t^q), will infuse \mathbf{Z}_a^q with information from \mathbf{z}_t^q . This task is expected to be difficult, due to the fact that some acoustic aspects may not be covered by the tags, or that some existing tags may be wrong. Therefore, we utilize two affine transforms, and we align the output of these transforms. Specifically, we utilize the affine transforms AFF_a and AFF_t , parameterized by $\theta_{\text{af-a}}$ and $\theta_{\text{af-t}}$, respectively, as

$$\Phi_a = \text{AFF}_a(\mathbf{Z}_a; \theta_{\text{af-a}}), \text{ and} \quad (5.8)$$

$$\phi_t = \text{AFF}_t(\mathbf{z}_t; \theta_{\text{af-t}}). \quad (5.9)$$

where $\Phi_a \in \mathbb{R}_{\geq 0}^{T' \times F}$ and $\phi_t \in \mathbb{R}_{\geq 0}^M$. Then, since Φ_a is a matrix and ϕ_t a vector, we flatten Φ_a to $\phi_a \in \mathbb{R}_{\geq 0}^{T'F'}$.

To align ϕ_a with its paired ϕ_t , we utilize randomly (and without repetition) sampled minibatches $\mathbb{G}_b = \{(\mathbf{X}_a^b, \mathbf{y}_t^b)\}_{b=1}^{N_b}$ from our dataset \mathbb{G} , where N_b is the amount of paired examples in the minibatch \mathbb{G}_b . For each minibatch \mathbb{G}_b , we align the ϕ_a^b with its paired ϕ_t^b and, at the same time, we optimize e_a , d_a , e_t , d_t , AFF_a and AFF_t . To do this, we follow (Chen et al., 2020) and we use the *NT-Xent* contrastive loss function

$$\ell_\xi(i, j) = -\log \frac{\Xi(\boldsymbol{\psi}^i, \boldsymbol{\psi}^j, \tau)}{\sum_{k=1}^{2N_b} \mathbb{1}_{[k \neq i]} \Xi(\boldsymbol{\psi}^i, \boldsymbol{\psi}^k, \tau)}, \text{ where} \quad (5.10)$$

$$\boldsymbol{\psi}^i = \begin{cases} \boldsymbol{\phi}_a^i & \text{if } i \leq N_b \\ \boldsymbol{\phi}_t^{i-N_b} & \text{if } i > N_b \end{cases} \quad (5.11)$$

$$\Xi(\mathbf{a}, \mathbf{b}, \tau) = \exp(\text{sim}(\mathbf{a}, \mathbf{b})\tau^{-1}), \quad (5.12)$$

$$\text{sim}(\mathbf{a}, \mathbf{b}) = \mathbf{a}^\top \mathbf{b} (\|\mathbf{a}\| \|\mathbf{b}\|)^{-1}, \quad (5.13)$$

$\Theta_c = \{\theta_{ea}, \theta_{af-a}, \theta_{et}, \theta_{af-t}\}$, $\mathbb{1}_A$ is the indicator function with $\mathbb{1}_A = 1$ iff A else 0, and τ is a temperature hyper-parameter. The final alignment loss $\mathcal{L}_\xi(\mathbb{G}_b, \Theta_c)$ is then calculated as the sum over all paired audio and tag representations, both (i, j) and (j, i) . Finally, we jointly optimize θ_{ea} , θ_{da} , θ_{et} , and θ_{dt} , for each minibatch \mathbb{G}_b , minimizing

$$\begin{aligned} \mathcal{L}_{\text{total}}(\mathbb{G}_b, \Theta) &= \lambda_a \sum_{b=1}^{N_B} \mathcal{L}_a(\mathbf{X}_a^b, \Theta_a) + \lambda_t \sum_{b=1}^{N_B} \mathcal{L}_t(\mathbf{y}_t^b, \Theta_t) \\ &\quad + \lambda_\xi \mathcal{L}_\xi(\mathbb{G}_b, \Theta_c), \end{aligned} \quad (5.14)$$

where $\Theta_a = \{\theta_{ea}, \theta_{da}\}$, $\Theta_t = \{\theta_{et}, \theta_{dt}\}$, Θ is the union of the Θ_* sets in Eq. (5.14), and λ_* is a hyper-parameter used for numerical balancing of the different learning signals/losses. After the minimization of $\mathcal{L}_{\text{total}}$, we use e_a as a pre-learned feature extractor for different audio classification tasks.

5.3 Experiment

We conduct an ablation study where we compare different methods for learning audio embeddings on their classification performance at different tasks, using as input the embeddings from the employed methods.

This allows us to evaluate the benefit of using the alignment and the reconstruction objectives in our method. We consider a traditional set of handcrafted features, as a low anchor. Additionally, we perform a correlation analysis with a set of acoustic features in order to understand what kind of acoustic properties are reflected in the learnt embeddings.

5.3.1 Pre-training dataset and data pre-processing

For creating our pre-training dataset \mathbb{G} , we collect all sounds from Free-sound that have a duration of maximum 10 seconds. We remove sounds that are used in any of our downstream tasks dataset. We apply a uniform sampling rate of 22 kHz and length of 10 secs to all collected sounds, by resampling and zero-padding as needed. We extract $F = 96$ log-scaled mel-band energies using sliding windows of 1024 samples (≈ 46 ms), with 50% overlap and the Hamming windowing function. We create overlapping patches of $T = 96$ feature vectors (≈ 2.2 s), using a step of 12 vectors for overlap. Then, we select the $T \times F$ patch with the maximum energy. This process is simple but we assume that in many cases, the associated tags will refer to salient events present in regions of high energy. We process the tags associated to the audio clips, by firstly removing any stop-words (obtained with the NLTK natural language processing Python library²) and making any plural forms of nouns (obtained with the inflect Python library³) to singular. We remove tags that occur in more than 70% of the sounds as they can be considered less informative, and consider the $C=1000$ remaining most occurring tags, which we encode using the multi-hot scheme. This corresponds to using the vector space model representation (Salton, 1989) and produces a feature that is a high-dimensional sparse vector where a value of 1 in one dimension refers to the presence of a specific tag. Finally, we discard sounds that were left with no tags after this filtering process. This process generated $Q = 189\,896$ spectrogram patches for our dataset \mathbb{G} . 10% of this patches are kept for validation and all the patches are scaled to values between 0 and 1.

²<https://www.nltk.org/>

³<https://github.com/jazzband/inflect>

We consider three different cases for our method for evaluating the benefit of the alignment and the reconstruction objectives. The first is the method presented in Section 5.2, termed as AE-C. At the second, termed as E-C, we do not employ d_a and d_t , and we optimize e_a using only \mathcal{L}_ξ , similar to (Chen et al., 2020). The third, termed as CNN, is composed of e_a , followed by two fully connected layers and is optimized for directly predicting the tag vector y_t using the CE function. Additionally, we employ the 20 first mel-frequency cepstral coefficients (MFCCs) with their Δ s and $\Delta\Delta$ s as a low anchor, using means and standard deviations through time, and we term this case as MFCCs. Finally, we compare the performance of our embeddings with different results taken from the literature. The models and results we compare to are presented later, in Section 5.3.4.

The code of our method is available online at: <https://github.com/xavierfav/coala>. We provide the pre-training dataset \mathbb{G} online, freely, and publicly at: <https://zenodo.org/record/3887261>. Sounds were accessed from the Freesound API on the 7th of May, 2019.

5.3.2 Utilized hyper-parameters, training procedure, and models

For the audio autoencoder, we use $N_{\text{CNN}}=5$ convolutional blocks each one containing $K_{l_{ea}} = 128$ filters of shape 4×4 , yielding an embedding ϕ_a of size 1152. This audio encoder model has approximately 2.4M parameters. The tag autoencoder is composed of $N_{\text{FNN}}=3$ layers of size 512, 512 and 1152, accepting a multi-hot vector of dimension 1000 as input. We train the models for 200 epochs using a minibatch size $N_B=128$, using an SGD optimizer with a learning rate value of 0.05. We utilize the validation set to define the different λ 's at Eq. (5.14) and the contrastive loss temperature parameter τ , to $\lambda_a=\lambda_t=5$, $\lambda_\xi=10$, and $\tau = 0.1$. We add a dropout regularization with rate 25% after each activation layer to avoid overfitting while training. The CNN baseline that is trained by predicting directly the multi-hot tag vectors from the audio spectrogram has follows the same architecture as the encoder from the audio autoencoder. It has

been trained for 20 epochs using a minibatch size $N_B=128$ and an SGD optimizer as well.

5.3.3 Downstream classification tasks

To assess the performance of the embeddings extracted with our embedding model e_a , we consider three different audio classification tasks:

- **Sound Event Recognition.** We use the UrbanSound8K dataset (US8K) (Salamon et al., 2014) in our experiment, which consists of around 8000 single-labeled sounds of maximum 4 seconds and 10 classes. We use the provided folds for cross-validation.

- **Music Genre Classification.** We use the fault-filtered version of the GTZAN dataset (Tzanetakis & Cook, 2002; Kereliuk et al., 2015) consisting of music excerpts of 30 seconds, single-labeled split in pre-computed sets of 443 songs for training and 290 for testing.

- **Musical Instrument Classification.** use the NSynth dataset (Engel et al., 2017) which consists of more than 300k sound samples organised in 10 instrument families. However, because we are interested to see how our models performs with relatively low amount of training data, we sample from NSynth a balanced set of 20k samples from the training set which correspond to approximately 7% of the original set. The evaluation set is kept the same.

For the above tasks and datasets, we use non-overlapping frames of audio clips that are calculated similarly to the pre-training dataset, and are given as input to the different methods in order to obtain the embeddings. Then, these embeddings are aggregated into a single vector (e.g. of 1152 dimensionality for our e_a) employing the mean statistic, and are used as an input to a classifier that is optimized for each corresponding task. Embeddings and MFCCs vectors are standardized to zero-mean and unit-variance, using statistics calculated from the training split of each task. As a classifier for each of the different tasks, we use a multi-layer perceptron (MLP) with one hidden layer of 256 features, similar to what is used in (Cramer et al., 2019). To obtain an unbiased evaluation of our

method, we repeat 10 times the training procedure of the MLP in each task, and average and report the mean accuracies.

5.3.4 Models from the literature

We compare the performance of our embedding in the different tasks and with different results taken from the literature. Here are presented the models and results that we then use as a comparison.

OpenL3 (Cramer et al., 2019) is an open source implementation of Look, Listen, and Learn (L3-Net) (Arandjelovic & Zisserman, 2017). It consists of an embedding model using blocks of convolutional and max-pooling layers, trained through self-supervised learning of audio-visual correspondence in videos from YouTube. The model has around 4.7M parameters and computes embedding vectors of size 6144. In Cramer et al. (2019), the authors report the classification accuracies of different variants of the model used as a feature extractor combined with a MLP classifier on the US8K dataset. Their mean accuracy is 78.2%.

VGGish (Hershey et al., 2017; Gemmeke et al., 2017) consists of an audio-based CNN model, a modified version of the VGGNet model (Simonyan & Zisserman, 2014) trained to predict video tags from the Youtube-8M dataset (Abu-El-Haija et al., 2016). The model has around 62M parameters and computes embedding vectors of size 128. Its accuracy when used as a feature extractor combined with a MLP classifier on the US8K dataset is reported in (Cramer et al., 2019) as being 73.4%.

DeepConv (Salamon & Bello, 2017) is a deep neural network composed of convolutional and max-pooling layers. When trained with data augmentation on the US8K dataset, it achieved 79.0% accuracy.

rVGG (Pons & Serra, 2019b) corresponds to a VGGish non-trained model (randomly weighted). The referenced work experiment using it as a feature extractor by comparing different embeddings from different layers of the network. The best accuracies on US8K and GTZAN when combined with an SVM classifier were reported as 70.7% and 59.7% respectively, using an embedding vector of size of 3585.

sampleCNN (Lee et al., 2018) is a deep neural network that takes as

input the raw waveform and is composed of many small 1D convolutional layers and that has been designed for musical classification tasks. When pre-trained on the Million Song Dataset (Bertin-Mahieux et al., 2011), this model reached a 82.1% accuracy on the GTZAN dataset.

smallCNN (Pons et al., 2017b) is a neural network composed of one CNN layer with filters of different sizes that can capture timbral characteristics of the sounds. It is combined with pooling operations and a fully-connected layer in order to predict labels. In (Ramires & Serra, 2019), it has been trained with the NSynth dataset in order to predict the instrument family classes and was reported to reach 73.8% accuracy.

5.3.5 Correlation analysis with acoustic features

We perform a correlation analysis using a similarity measure involving the Canonical Correlation Analysis (CCA) (Hardoon et al., 2004), to investigate the correlation of the output embeddings from our method, with various low-level acoustic features. Similar to (Raghu et al., 2017), we use sounds from the validation set of the pre-training dataset \mathbb{G} , and we compute the canonical correlation similarity of our audio embedding \mathbf{Z}_a with statistics of acoustic features computed with the librosa library (McFee et al., 2015). These features correspond to MFCCs, chromagram, spectral centroid, and spectral bandwidth, all computed at a frame level.

5.4 Results

5.4.1 Classification performance

In Table 5.1 are the results of the performance of the different embeddings and our MFCCs baseline, and results reported in the literature. In all tasks, all the learned embeddings yielded better results than the MFCCs baseline, showing that it is possible to learn meaningful audio representations, by taking advantage of tag metadata. However, the CNN case does not even reach the performance of the MFCCs features. This clearly indicates the benefit of our approach for building general audio representations by

Table 5.1: Average mean accuracies for SER, MGC, and MIC. Additional performances are taken from the literature (Cramer et al., 2019; Salamon & Bello, 2017; Pons & Serra, 2019b; Lee et al., 2018; Ramires & Serra, 2019).

	US8K	GTZAN	NSynth
MFCCs	65.8	49.8	62.6
AE-C	72.7	60.7	73.1
E-C	72.5	58.9	69.5
CNN	48.4	47.0	56.4
OpenL3	78.2	–	–
VGGish	73.4	–	–
DeepConv	79.0	–	–
rVGG	70.7	59.7	–
sampleCNN	–	82.1	–
smallCNN	–	–	73.8

leveraging user-provided noisy tags . When comparing the different proposed embeddings, we see that the AE-C case consistently leads to better results. For the MIC (NSynth) task, combining reconstruction and contrastive objectives (i.e. AE-C case) brings considerable benefit. For the MGC (GTZAN) task, these benefits are still significant but not as great, and finally, when looking at the SER (US8K) task, adding the reconstruction objective does not improve the results much. Our assumption is that recognizing musical instruments can be more easily done using lower-level features reflecting acoustic characteristics of the sounds, and that the reconstruction objective imposed by the autoencoder architecture is forcing the embedding to reflect low-level characteristics present in the spectrogram. However, for recognizing urban sounds or musical genres, a feature that reflects mainly semantic information is needed, which seems to be learned successfully when considering the contrastive objective.

Comparing our method to others for the SER, we can see that we are slightly outperformed by VGGish (Hershey et al., 2017; Gemmeke et al., 2017), according to results taken from (Cramer et al., 2019), which has been trained with million of manually annotated audio files using pre-

defined categories. This shows that our approach which only takes advantage of small-scale content with their original tag metadata is very promising for learning competitive audio features. However, our model is still far from reaching performances given by OpenL3 or the current state-of-the-art DeepConv with data augmentation. Similarly in MGC, using a sampleCNN as embedding model trained on the Million Song Dataset (MSD) (Lee et al., 2018) produces better results than our approach. But, all these models have been either trained with much more data than our approach, or use a more powerful classifier than we used. Finally, NSynth dataset has been originally released in order to train generative models rather than classifiers. Still, results from (Ramires & Serra, 2019), show that our approach training using around 8% of the training data, is only slightly outperformed by a CNN trained with all the training data (small-CNN).

5.4.2 Correlation analysis

Table 5.2 shows the correlation for the different embeddings Z_a with the mean, the variance, and the skewness of the different acoustic feature vectors. Overall, we observe a consistent increase of the correlation between the acoustic features and embeddings trained with models containing an AE structure. This suggests that the reconstruction objective enables to learn features that reflect some low-level acoustic characteristics of audio signals, which makes it more valuable as a general-purpose feature. More specifically, there is a significant correlation increase between the MFCCs mean and models that contain AE structure, showing that they can capture more timbral characteristics of the signal. However, variance and skewness did not significantly increase, which shows that our embeddings lack to capture temporal cues. Considering chromagrams, which reflect the harmonic contents of a sound, we see little improvement with AE models. This suggests that our embeddings still lack some important musical characteristics. Regarding the lower-level features spectral centroid and bandwidth, we only observe a slight increase of correlations with AE-based embeddings. This suggests that spectrogram-based CNN

Table 5.2: CCA correlation scores between the embeddings model outputs and some acoustic features statistics.

	mean	var	skew	mean	var	skew
	MFCCs			Chromagram		
AE-C	0.84	0.51	0.42	0.48	0.37	0.40
E-C	0.58	0.49	0.39	0.38	0.36	0.32
CNN	0.73	0.43	0.32	0.59	0.33	0.48
	Spectral Centroid			Spectral Bandwidth		
AE-C	0.97	0.87	0.80	0.96	0.86	0.84
E-C	0.93	0.82	0.76	0.92	0.82	0.81
CNN	0.95	0.76	0.74	0.91	0.72	0.80

models can already reflect some low-level acoustic characteristics of the signals, without the need of involving a reconstruction objective during the training.

5.4.3 Clustering performance

As we observe indications that the features learned combining autoencoders with the contrastive learning can be performed competitively with state-of-the-art features in the case of supervised classification tasks with relatively small datasets, we are now interested in seeing how well these features can perform in a clustering scenario. One of our main goals is to learn a representation that reflects both low-level and high-level characteristics for a wide range of everyday sounds, which can be adequate when having to cluster sounds from large and varied sound collections. In this experiment, we perform an external validation using the same collection of datasets used for the evaluation of the clustering approaches (Section 4.4.1). However, because both the training dataset and the ones used for the clustering evaluation contain data from Freesound, approximately 70% of the sounds in the evaluation set were present in the dataset for training the model. Therefore, we remove from the evaluation datasets these sounds which results in having much less data for the eval-

uation. Although this is not an ideal evaluation setup, it gives us an idea of the potential of our learned representations for the unsupervised classification of sounds. Additionally, we decide to remove the evaluation of the SoundNet features, given the poor performance they showed in Chapter 4.

We compare clustering performances using two different approaches (K-means and a k-NN graph) on many small datasets. The results which are shown in Table 5.3, suggest that our embedding can produce good performance for clustering tasks, sometimes better than state-of-the-art models. One intuition on why the feature we learn seem to produce better result is that they may be more suited for clustering methods that employ a similarity notion in the embedding space. Since our approach involves a contrastive loss that relies on cosine similarities in the embedding space, it produces a feature space that can be more naturally used by our clustering methods. For the embeddings taken from the literature, the objective function comes from either a binary cross-entropy loss in the case of AudioSet when predicting labels, or as a modality matching prediction in the case of OpenL3. This produces good discriminative features, but may be less adapted to computing similarities in the embedding latent space.

5.5 Conclusions

In this work we present a method for learning an audio representation that can capture acoustic and semantic characteristics for a wide range of sounds. We utilise two heterogeneous autoencoders (AEs), one taking as an input audio spectrogram and the other processing a tag representation. These AEs are jointly trained and a contrastive loss enables to align their latent representations by leveraging associated pairs of audio and tags. We evaluate our method by conducting an ablation study, where we compare different methods for learning audio representations over three different classification tasks. We also perform a correlation analysis with acoustic features in order to grasp knowledge about what type of acoustic characteristics the embedding captures. And we finally evaluate our best learned embedding in the context of clustering.

Dataset	MFCC		AudioSet		OpenL3 music		OpenL3 env		AE-C	
	K-means	KNN	K-means	KNN	K-means	KNN	K-means	KNN	K-means	KNN
Sound of things	0.150	0.096	0.206	0.188	0.106	0.100	0.173	0.137	0.149	0.213
Natural Sounds	0.358	0.350	0.417	0.459	0.354	0.358	0.395	0.349	0.419	0.379
Vehicles	0.120	0.081	0.125	0.117	0.025	0.064	0.029	0.045	0.097	0.100
Instruments	0.136	0.144	0.168	0.181	0.182	0.198	0.153	0.195	0.173	0.227
Animals	0.349	0.189	0.342	0.167	0.140	0.150	0.165	0.138	0.216	0.218
Human Sounds	0.136	0.108	0.262	0.210	0.138	0.149	0.182	0.204	0.162	0.281
Average	0.208	0.161	0.253	0.220	0.158	0.170	0.183	0.178	0.203	0.236

Table 5.3: Average performance (AMI) across the different dataset families of the K-means and the KNN-Graph clustering methods with the different features. An AMI close to 0 corresponds to a random partition while perfect matches gives 1 AMI.

Results indicate that combining reconstruction objectives with a contrastive learning framework enables to learn audio features that reflect both semantic and lower-level acoustic characteristics of sounds, which makes it suitable for general audio machine listening applications. Also, our learned representations seem to produce better results for the unsupervised classification of sounds.

Chapter 6

Learning Contextual Tag Embeddings for Cross-modal Alignment of Audio and Tags

Self-supervised audio representation learning offers an attractive alternative for obtaining generic audio embeddings, capable to be employed into various downstream tasks. In the last chapter, we investigated the use of audio and associated tags in order to learn audio features. However, the text-based processing model we used is not capable to generalize to tags that were unseen during training. In this chapter we propose a method for learning audio representations using an audio autoencoder (AAE), a word embedding model (WEM), and a multi-head self-attention (MHA) mechanism. MHA attends on the output of the WEM, providing a contextualized representation of the tags associated with the audio, and we align the output of MHA with the output of the encoder of AAE using a contrastive loss. We jointly optimize AAE and MHA and we evaluate the audio representations (i.e. the output of the encoder of AAE) by utilizing them in three different downstream tasks, namely sound, music genre, and music instrument classification.

6.1 Introduction

In the natural language and image processing fields, both supervised and unsupervised approaches enabled the creation of powerful pre-trained models, that are often employed in many different tasks (Mikolov et al., 2013; Devlin et al., 2018; Chen et al., 2020). The association of images and text can be exploited for learning embeddings (Wu et al., 2019), e.g. by using a self-attention mechanism to learn context sensitive text embeddings that are then aggregated into sentence embeddings. Similar approaches have been adopted for machine listening, for instance unsupervised pre-training of transformer models can improve speech recognition performance by employing contrastive predictive coding strategies (et al., 2019; Oord et al., 2018). In (Turpault et al., 2019b), the authors employ a semi-supervised sampling strategy to create triplets for benefiting automatic tagging systems. However, these approaches consider just one modality, i.e. audio.

A cross-modal method for learning and aligning audio and tag latent

representations (COALA) was presented in the last chapter and in Favory et al. (2020a). Latent representations were learnt using autoencoders, one aiming at encoding and reconstructing spectrogram representations of sounds, and the other focusing on encoding and reconstructing a set of tags represented as multi-hot vectors. The outcomes suggest that it is possible to leverage user-generated audio data and accompanying tags, for learning semantically enriched audio representations that can be used for different classification tasks. However, in COALA the tag-based input representation was fixed, and therefore the tag-based encoder cannot generalize to new terms that have not been seen during training. This makes the approach lose the flexibility of contrastive representation learning.

In this chapter we propose a method for allowing the textual generalization of cross-modal approaches, using pre-trained word embedding models which project words into semantic spaces. We propose to use an attention mechanism for learning higher-level contextualized semantic representation similarly to Wu et al. (2019). However, our approach relies on accompanying tags instead of text, and therefore employs a simpler approach for computing semantic embeddings. The rest of the chapter is as follows. In Section 2 we present our proposed method. Section 3 describes the utilized dataset, the tasks and metrics that we employed for the assessment of the performance, the baselines that we compare our method with. The results of the evaluation is presented and discussed in Section 4. Finally, Section 5 concludes the chapter and proposes future research directions.

6.2 Proposed method

Our method, illustrated in Figure 1, consists of an audio encoder, $e_a(\cdot)$, an audio decoder, $d_a(\cdot)$, a pre-trained word embedding model, $e_w(\cdot)$, and multi-head self-attention, $\text{Att}(\cdot)$. As input to our method, we employ a dataset of N_B paired examples, $\mathbb{G} = \{(\mathbf{X}_a^{n_B}, \mathbf{x}_w^{n_B})\}_{n_B=1}^{N_B}$, where $\mathbf{X}_a \in \mathbb{R}^{T_a \times F_a}$ is a time-frequency audio representation of T_a audio feature vectors of F_a features, and $\mathbf{x}_w = \{x_w^i\}_{i=1}^{T_w}$ is a set of T_w tags, x_w^i , like “techno”

and “bark”, and associated with \mathbf{X}_a ¹. Att extracts from \mathbf{x}_w an embedding containing the context of the tags, and by using a contrastive loss we align the output $\mathbf{z}_a = e_a(\mathbf{X}_a)$ with the output of Att. \mathbf{z}_a is also used as an input to d_a , to reconstruct \mathbf{X}_a , effectively infusing \mathbf{z}_a with both semantic (from \mathbf{x}_w) and low-level acoustic information (from the reconstruction by d_a). The code of our method is available online².

6.2.1 Audio encoding and decoding

As in COALA, presented in chapter 5, the encoder e_a consists of N_{e-a} cascaded 2D convolutional neural networks (CNNs), $\text{CNN}_{n_{e-a}}$, with $C_{n_{e-a}}^{\text{in}}$ and $C_{n_{e-a}}^{\text{out}}$ input and output channels, respectively, a square kernel of K_{e-a} size, and S_{e-a} stride. The CNNs process \mathbf{X}_a in a serial fashion, and each $\text{CNN}_{n_{e-a}}$ is followed by a batch normalization process ($\text{BN}_{n_{e-a}}$), a rectified linear unit (ReLU), and a dropout (DO) with probability p_a , as

$$\mathbf{H}_{n_{e-a}} = \text{DO}(\text{ReLU}(\text{BN}_{n_{e-a}}(\text{CNN}_{n_{e-a}}(\mathbf{H}_{n_{e-a}-1}))))), \quad (6.1)$$

where $\mathbf{H}_0 = \mathbf{X}_a$. $\mathbf{H}_{N_{e-a}} \in \mathbb{R}_{\geq 0}^{C_{N_{e-a}}^{\text{e-out}} \times T'_{N_{e-a}} \times F'_{N_{e-a}}}$ is flattened to a vector and given as an input to a layer normalization process (LN) (Ba et al., 2016) (FFN_{e-a}), as $\mathbf{z}_a = \text{LN}(\mathbf{h}_{N_{e-a}})$, where $\mathbf{h}_{N_{e-a}}$ is the flattened $\mathbf{H}_{N_{e-a}}$, and $\mathbf{z}_a \in \mathbb{R}^V$, with $V = C_{N_{e-a}}^{\text{e-out}} \cdot T'_{N_{e-a}} \cdot F'_{N_{e-a}}$, is the learned audio embedding by our method. \mathbf{z}_a is used at the employed contrastive loss, in order to be aligned with the information contained at the associated tags \mathbf{x}_w , and as an input to d_a in order to encode low-level acoustic features in \mathbf{z}_a .

The decoder d_a takes as an input \mathbf{z}_a and processes it through a series of N_{e-a} transposed 2D CNNs (Radford et al., 2016; Dumoulin & Visin, 2016), $\text{CNN}_{n_{d-a}}$, in a reverse fashion to e_a . Firstly \mathbf{z}_a is turned to the matrix $\mathbf{Z}_a \in \mathbb{R}^{C_{N_{e-a}}^{\text{e-out}} \times T'_{N_{e-a}} \times F'_{N_{e-a}}}$ and then is processed by $\text{CNN}_{n_{d-a}}$ as

$$\mathbf{H}_{n_{d-a}} = \text{ReLU}(\text{BN}_{n_{e-a}}(\text{CNN}_{n_{d-a}}(\text{DO}(\mathbf{H}_{n_{d-a}-1}))))), \quad (6.2)$$

¹For the clarity of notation, the superscript n_B is dropped here and for the rest of the document, unless it is explicitly needed.

²<https://github.com/xavierfav/ae-w2v-attention>

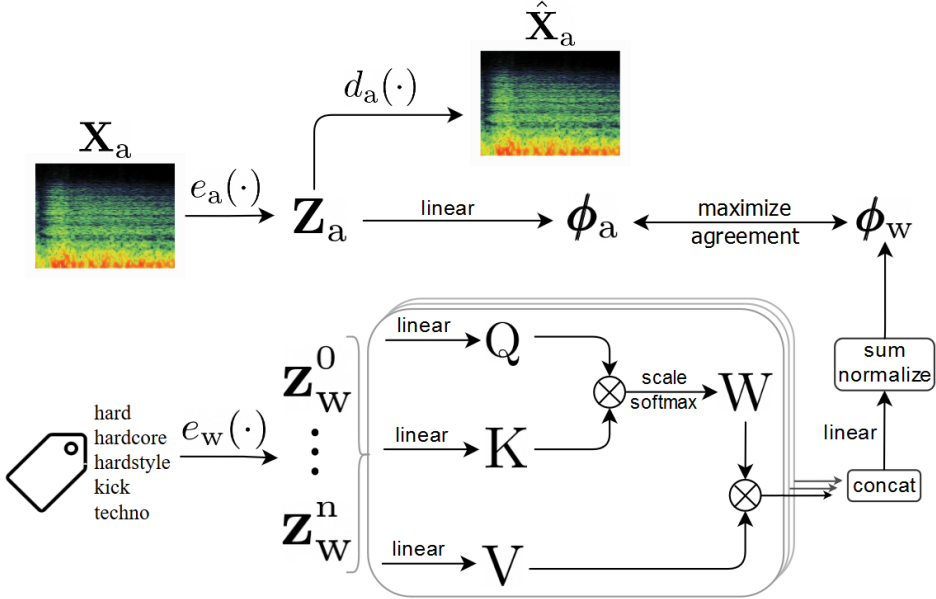


Figure 6.1: Illustration of our method. ϕ_a and ϕ_w are aligned by maximizing their agreement through contrastive learning and, at the same time, Z_a is used for reconstructing back the original spectrogram input. Word embeddings are passed through a multi-head scaled dot-product self-attention layer in order to build higher-level semantic vectors that are finally aggregated into a single vector ϕ_w .

where $H_0 = Z_a$ and $BN_{n_{e-a}}$ is the batch normalization process used after $CNN_{n_{d-a}}$. The reconstructed input, \hat{X}_a , is obtained as $\hat{X}_a = \sigma(H_{N_{d-a}})$, where σ is the sigmoid function.

6.2.2 Multi-head, self-attention tags encoding

As our e_w we select a pre-optimized word embedding model, using an embedding dimensionality of F_w , which outputs $z_w^{t_w} \in \mathbb{R}^{F_w}$. For processing the output of e_w we follow the recent proposal of multi-head, self-attention in the Transformer model, where a scaled dot-product attention

mechanism is employed to extract the relevant information of each word in a sentence (et al., 2017). Since we use an unordered set of tags, we do not employ the positional embeddings for each tag. The multi-head self-attention attends on the set of encoded tags by the word embedding model e_w , and extracts a contextual embedding of the set of tags. We use this contextual embedding as the latent representation of tags for our cross-modal alignment process with the contrastive loss.

Specifically, we employ three feed-forward neural networks, FNN_q , FNN_k , and FNN_v , FNN_o (without non-linearities) as our linear transformations for the query, key, value and output of the self-attention mechanism, respectively. We follow the original paper of the self-attention (et al., 2017), we use H attention heads, and we apply the self-attention, Att , on the embeddings of tags, \mathbf{Z}_w . The output of Att is $\mathbf{O} \in \mathbb{R}^{H \times T_w \times F_w}$, according to (et al., 2017). Then, we concatenate the result of each H , resulting to $\mathbf{O}' \in \mathbb{R}^{(H \cdot T_w) \times F_w}$. Finally, to process the output of each attention head H and obtain the contextual embedding for the input tags, ϕ_w , we employ FNN_o and a layer normalization process, LN , and we calculate the contextual embedding, $\phi_w \in \mathbb{R}^V$, as

$$\mathbf{O}' = \text{FNN}_o(\mathbf{O}) \text{ and} \quad (6.3)$$

$$\phi_w = \text{LN}\left(\sum_{i=1}^{T_w} \mathbf{O}'_i\right), \quad (6.4)$$

where $\mathbf{O}' \in \mathbb{R}^{T_w \times V}$ and FNN_o has its weights shared along the $H \cdot T_w$ dimension.

6.2.3 Cross-modal alignment and optimization

To align the learned latent representations from audio and tags, we employ a constrastive loss and we maximize the agreement of ϕ_a^{nB} and ϕ_w^{nB} on minibatches \mathbb{G}_b of size N_b , randomly sampled from our dataset \mathbb{G} . To reduce the mismatch between the two different modalities, we employ a feed-forward neural network (FNN_{c-a}) to process \mathbf{z}_a^{nB} , as

$$\phi_a^{nB} = \text{FNN}_{c-a}(\mathbf{z}_a^{nB}). \quad (6.5)$$

Then, we employ a contrastive loss between ϕ_a^{nB} and ϕ_w^{nB} and we jointly optimize e_a , d_a , FNN_q , FNN_k , FNN_v , and FNN_o by minimizing the loss

$$\mathcal{L}(\mathbb{G}_b, e_a, d_a, \text{Att}) = \lambda_a \mathcal{L}_a + \lambda_\xi \mathcal{L}_\xi, \quad (6.6)$$

where \mathcal{L}_a is the generalized Kullback-Leibler divergence between \mathbf{X}_a and $\hat{\mathbf{X}}_a$, shown to work good for audio reconstruction (Drossos et al., 2018; Mimitakis et al., 2018). \mathcal{L}_ξ is the contrastive loss between paired ϕ_a^{nB} and ϕ_w^{nB} and other examples in the minibatch, as defined in (Chen et al., 2020). We use a temperature parameter τ for \mathcal{L}_ξ , and λ_\star is a hyper-parameter used for numerical balancing of the two losses.

6.3 Evaluation

We evaluate our method by assessing the performance of e_a as a pre-trained audio embedding extractor in different audio classification tasks. We utilize a different audio dataset for each task and we compare the performance of our e_a against COALA and a set of handcrafted MFCCs feature.

6.3.1 Pre-training dataset and data pre-processing

We use the same dataset as used in COALA, which consists of sounds and associated tags collected from Freesound platform (Font et al., 2013a). We compute $F_a = 96$ log-scaled mel-band energies using sliding windows of 1024 samples (≈ 46 ms), with 50% overlap and the Hamming windowing function. Then, we select the spectrogram patch of size $T_a = 96$ that has maximum energy in each sample. This process leads to 189 896 spectrogram patches. 10% of these patches are kept for validation and all the patches are scaled to values between 0 and 1. We process the tags associated to the audio clips by firstly removing any stop-words and making any plural forms of nouns to singular. We remove tags that occur in more than 70% of the sounds as they can be considered less informative, and consider the $C=1000$ remaining most occurring tags. Then we

train a continuous bag-of-words Word2Vec model (Mikolov et al., 2013) using these processed tags and we use this model as our e_w .

To have our method comparable with COALA and assess the impact of our proposed approach for learning contextual tags, we follow COALA and we use $N_{e-a} = 5$ CNN $_{n_{e-a}}$, with $K_{e-a} = 4$ and $S_{e-a} = 2$. We use $C_1^{in} = 1$ and $C_{n_{e-a}}^{out} = 128$. These hyper-parameters result to $V = 1152$ and e_a has approximately 2.4M parameters. The tag encoder takes a set of $T_w = 10$ maximum tags. It uses fully-connected linear transformations that retain the same dimension as the word embeddings, producing tag-based embedding vectors of the same dimension. We utilize two different F_w , one of 128 and another of 1152. The first of 128 is due to the fact that we are using a small scale vocabulary and we choose to have a small embedding size for e_w . The second, is for having $F_w = V$. Additionally, we employ two different set-ups for our Att, one with far and another with one attention head. We indicate the different combinations as “w2v- F_w -Hh”, where H is the amount of attention heads. For example, “w2v-128-1h” means that we are using $F_w = 128$ and one attention head. We also employ the self-attention strategy with a simple mean aggregation of the tags’ word embeddings, which we refer as w2v-128-mean and w2v-1152-mean, to assess the impact of Att when using a simpler aggregation of its output. Finally, we employ the 20 first mel-frequency cepstral coefficients (MFCCs) with their Δ s and $\Delta\Delta$ s as a low anchor, using means and standard deviations through time, and we term this case as MFCCs.

All our models are trained for 200 epochs, using a minibatch size $N_B=128$ and an SGD optimizer with a learning rate value of 0.005. We utilize the validation set to define the different λ ’s at Eq. (11) and the contrastive loss temperature parameter τ , to $\lambda_a=5$, $\lambda_\xi=10$, and $\tau = 0.1$. We use dropout probability of 0.25 for our e_a and d_a and 0.1 after the tag-based embedding model to avoid overfitting while training.

6.3.2 Audio-based classification

We assess the performance of the different embeddings extracted with e_a in three different audio classification tasks. For each task, we employ

the pre-trained, with our method, e_a , and we extract audio embeddings z_a for the the audio data of the corresponding task. Then, adopt the corresponding training protocol of the task (e.g. cross-fold validation) and we optimize a multi-layer perceptron (MLP) with one hidden layer of 256 features, similar to what is used in (Cramer et al., 2019; Favory et al., 2020a). Finally, we assess the performance of the classifier using the data and the testing protocol of each task. To obtain an unbiased evaluation of our method, we repeat 10 times the training procedure of the MLP in each task and report the mean accuracy. Additionally, in order to understand if the self-attention mechanism is actually beneficial for obtaining a better semantic embedding from tags, we perform a tag-based classification for comparing the different approach versions. For this purpose, we use the UrbanSound8K dataset and the tags of the associated samples from Freesound.

As done in the evaluation proposed in Chapter 5, we first consider Sound Event Recognition (SER), where we use the UrbanSound8K dataset (US8K) (Salamon et al., 2014). We additionally consider the task of Music Genre Classification (MGC), where we use the fault-filtered version of the GTZAN dataset (Tzanetakis & Cook, 2002; Kereliuk et al., 2015). Finally, we consider the Musical Instrument Classification (MIC) task. We use the same sample of the NSynth dataset used in the last chapter (Engel et al., 2017).

For the above tasks and datasets, we use non-overlapping audio frames that are calculated similarly to the pre-training dataset. These frames are given as input to the different models in order to obtain audio embeddings. In order to obtain fixed-length vectors, the audio embeddings are aggregated using a mean average statistic and finally used as an input to a classifier that is trained for each corresponding task. Additionally, resulting embedding and MFCCs vectors are standardized to zero-mean and unit-variance, using statistics calculated from the training split of each task.

6.4 Results

Table 1 shows the performances of the different embeddings, our MFCCs baseline and previous results from COALA (Chapter 5). The self-attention mechanism used in the tag-based network benefits the classification performance in SER and MGC. This indicates that our proposed method indeed results in learning a contextual embedding that can be effectively used for learning better general audio representations. For MIC however, we do not observe any benefit from it. A reason may be that the classification of a musical instrument does not rely much on the context of employed textual descriptions, at least for the musical instruments contained in the employed dataset for MIC. This means that that classifying instrument samples can be done by solely using representations learned by the audio autoencoder, without any semantic information. In the previous chapter, it was observed that the reconstruction objective was bringing important improvements in this case, and probably, the enrichment of semantics achieved with the alignment with the tag-based latent representation loses its benefits. Using more attention heads is able to bring better performance in SER and MGC. This suggests that the pre-trained word representation we use can benefit from a more powerful attention mechanism. The impact of the embedding size of the word embeddings is not clear from our experiment. But, our findings suggest that using different dimensions for the audio autoencoder and the tag-based encoder does not necessarily hinder the contrastive alignment.

When using tags for performing the classification on US8K for SER, there is no benefit of using multiple attention heads, and the self-attention mechanism is only slightly improving the performance compared to the mean aggregation strategy. Moreover, the results show that audio-based classification is still not performing as well as the tag-based one, which could suggest that more powerful audio encoders could be better aligned with the semantics of the content, and produce better results.

Table 6.1: Average mean accuracy for SER, MGC, and MIC. Additionally performances on US8K dataset using a tag-based classifier are reported in the last column.

	SER	MGC	MIC	US8K-tag
MFCCs	65.8	49.8	62.6	-
w2v-128-1h	71.5	61.3	68.9	79.2
w2v-1152-1h	72.1	61.5	68.6	80.3
w2v-128-4h	73.5	59.6	69.7	79.4
w2v-1152-4h	70.5	63.4	69.9	78.7
w2v-128-mean	71.3	60.4	70.0	79.7
w2v-1152-mean	71.1	60.7	68.4	78.5
COALA	72.7	60.7	73.1	-

6.5 Conclusion

In this chapter we presented a method for cross-modal alignment of audio and tags. The proposed approach uses a pre-trained word embedding and learns contextual tag embeddings that are aligned with audio embeddings using contrastive learning. From audio samples and associated tags, our method is able to learn semantically enriched audio representation that can be used in different classification tasks. The embedding model produced is evaluated in three different downstream tasks, including sound event recognition, music genre and music instrument classifications. Over the previous similar method COALA, presented in Chapter 5, the proposed approach relies on pre-trained word embeddings that grants the the advantage of being directly able to be used in a wider range of applications, such as cross-modal retrieval or zero-shot learning.

However, in order to enable these types of applications, further investigations need to be performed. For instance, considering a more general pre-trained word embedding model, trained with a larger vocabulary, would be interesting. For that, leveraging additional text information, such as the accompanying text description of the sounds in Freesound, or textual descriptions of audio categories from AudioSet (Gemmeke et al., 2017) could enable a better generalization towards natural language. For

cross-modal retrieval, future work could focus in evaluating our approach in this context. Similar approaches in the video processing field indicate promising results in this type of application (Galanopoulos & Mezaris, 2020)

Chapter 7

Search Results Clustering

The large size of nowadays' online multimedia databases makes retrieving their content a difficult and time-consuming task. Users of online sound collections typically submit search queries that express a broad intent, often making the system return large and unmanageable sets of results. Search Result Clustering (SRC) is a technique that organises search result contents into coherent groups, which allows users to identify useful subsets in their results. Obtaining coherent and distinctive clusters that can be explored with a suitable interface is crucial for making this technique a useful complement of traditional search engines.

In this chapter, we present a SRC system that we developed and integrated in the Freesound website. The clustering algorithm relies on the graph-based approach presented in Chapter 4, which can be used with different audio features as input. One important requirement is that the algorithm needs to be able to produce acceptable performance on the wide variety of types of content present in online sound collections such as Freesound. We propose an approach to assess the performance of different audio features at scale, by taking advantage of the metadata associated with each sound. This analysis is complemented with an evaluation using ground-truth labels from manually annotated datasets, similarly to what has been presented in Chapter 4. We also show that using a confidence measure for discarding inconsistent clusters improves the quality of the partitions according to the available ground truth labels. After identifying the most appropriate features for clustering, we conducted two experiments with users performing a sound design and a music related task respectively. This allowed us to evaluate our approach and its user interface. Such experiments were followed by usability questionnaires and semi-structured interviews with the participants. This provided us with valuable novel insights regarding the features and specifications that promote efficient interaction with the clusters.

7.1 Introduction

Sounds used in movies, video-games, music and other media often originate from sound collections. Given the often large size of sound collections, retrieving content effectively from them can become challenging. Searching content in collaborative collections is further hindered by the heterogeneity of the content metadata.

The primary role of retrieval systems is to support users in accessing relevant content according to their needs. The relevance of results depends significantly on the specific use-case. Most individuals experience similar problems when for instance exploring music collections. However, in this case, people are generally interested in retrieving or discovering content that fits their taste or current mood. This media consumption corresponds to what has been referred as the *read-only* culture, which is in fact the dominant approach in modern mass-media culture (Lawrence, 2008). However, when browsing sound collections for creative purposes, such as sound design or music creation, the pertinence of the content depends on the specific user task. In contrast with the *read-only* approach, in a *read-write* culture (also known as the *remix* culture), individuals combine, rearrange and edit existing materials in a creative way to produce new content.

When interacting with sound collections, users typically rely on text-based search engines. After entering a text query, a user often faces a long list of results. In the absence of specific query terms, the system may be unable to differentiate the relevance of the retrieved sounds to the user, whose needs are frequently very precise and highly specialised. The user might be looking for audio clips, with distinctive and detailed characteristics, that can be described by a wide range of properties. In sound design, for instance, a user could be searching for a door-closing sound with a grinding noise that fits the movie ambiance and the visual aspect of the door. In the case of music creation, they may be interested in finding instrument loops in a certain tonality, at a specific tempo and with different timbres. In order to locate sounds of interest, the user usually needs to inspect the results one by one, listening to some of them

and judging their relevance. The process of finding the most appropriate sounds can be very time consuming and fail to retrieve important sounds, when interacting with large collections.

Users can narrow down the search by reformulating their queries by using for instance facet-based and tag-based filters (Tunkelang, 2009). When available, a user might find it very informative to explore the text and tags accompanying the sounds in the results page. This can help identify new terms for reformulating the query. Tag clouds that organise and display popular tags from the results are particularly useful for that purpose. The user can quickly find particular sub-topics in the search results (Sinclair & Cardew-Hall, 2008). Nonetheless, functionalities based on textual metadata depend critically on the quality of the annotations, which is often limited in collaborative collections. For this reason, content-oriented methods that are based on the audio content itself, have increased potential in the development of novel approaches to navigate search results.

To that end, one complementary feature that search engines can incorporate is audio-based SRC, which consists of grouping the results into labeled clusters or categories. It allows the user to submit a weakly-specified query and then explore the different themes that have been automatically extracted from the query results. Clustering engines can complement the search by providing a faster way to retrieve relevant items, facilitating topic exploration and preventing the overlook of information (Carpineto et al., 2009). However, such systems depend on more than just the clustering algorithm. In order to guide the user to locate relevant items in the different clusters, meaningful labels should be assigned to each of them. Moreover, the clustering is desirable to be performed online within a short response time, therefore requiring high computational efficiency. Finally, the clustering engine requires a graphical user interface that provides an intuitive way to navigate the clusters, e.g. by conveying visual information.

SRC has been extensively studied in the context of web documents (Carpineto et al., 2009; Zamir & Etzioni, 1999; Osiński, 2003; Mecca et al., 2007; Sadaf & Alam, 2012) and images (Cai et al., 2004; Jing et al.,

2006; Ben-Haim et al., 2006). Although many web search engines incorporating clustering do not exist anymore, Carrot2 (Stefanowski & Weiss, 2003) is probably one of the most popular remaining ones. It is open source and still in active development. It can automatically cluster small collections of web documents. It consists of a software component that can be integrated with other software as an external component (with the usage of an API). Several clustering algorithms are available, as well as different ways to interact with the clusters. For instance, the clusters can be displayed as a list of folders which content can be individually inspected as seen in Figure 7.1. Alternative cluster visualisations include a Voronoi treemap (Balzer & Deussen, 2005) and pie-chart.

Another popular commercial web clustering engines is Yippy¹ (formerly Clusty). It is probably the tool that is applied to the biggest size of web indexed document available publicly. When submitting a query to the search engine, many results are retrieved from the web. A list of clusters can be explored in the same way we explore folders. Moreover, it includes a certain hierarchy in the obtained clusters (Figure 7.2).

In the case of images, Google proposes a tool that can identify different groups from the search results (Figure 7.3). A list of labels appear at the top of the results, which allows to filter the results. Interestingly, when adding one of the labels, the cluster labels get updated and provide new ones that can correspond to more specific topics covered in the previously selected cluster. The clustering method is not known, but it may probably be similar to what is present by (Liu et al., 2007), which relies on approximate nearest neighbor searchers for scalability and uses content-based (image) features.

In the remaining of this chapter, we describe the development of an audio-based SRC engine that can be integrated in the Freesound website (Font et al., 2013a). In Section 2, we provide an overview of related work. We then introduce our graph-based clustering approach and our interface in Section 3. In Section 4, we compare the performance of different features taken from the literature, by using sound metadata and ground-truth labels from manually-annotated datasets. Section 5 presents

¹<https://yippy.com/>

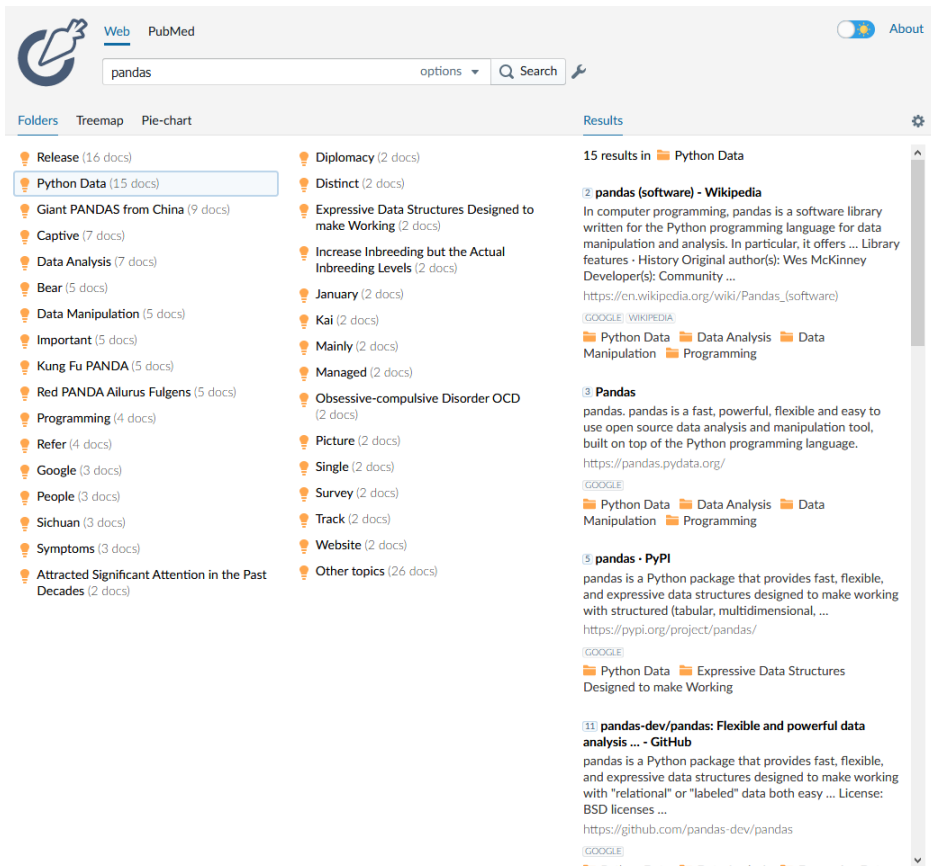


Figure 7.1: Screenshot of the publicly available web search results clustering interface using Carrot2 requested with the “pandas” query.

The screenshot shows the Yippy search engine interface. At the top left is the Yippy logo. To its right are links for 'Web', 'News', 'Images', and 'Video'. Below these is a search bar containing the text 'pandas' and a green 'Search' button. A grey bar below the search bar indicates 'Results 1-18 of 18 in Python, Data'. On the left side, there is a sidebar with 'Sources Sites Time Topics' and a list of categories including 'Top 318 Clusters', 'Bears (42)', 'Games (43)', 'Coins, Silver (26)', 'Pictures (27)', and 'Python, Data (18)'. The 'Python, Data (18)' category is expanded, showing sub-categories like 'Pandas Tutorial (4)', 'Stack Overflow (3)', 'Pandas Is A Python Package That Provides Fast, Flexible (2)', 'SciPy, IPython (2)', and 'Other Topics (7)'. The main content area displays six search results for 'pandas', each with a title, date, and a brief description. The results are: 1. 'pandas - Python Data Analysis Library', 2. 'Plot With Pandas: Python Data Visualization for Beginners...', 3. 'Data Analysis Made Simple: Python Pandas Tutorial', 4. 'Python Pandas - Introduction - Tutorialspoint', 5. 'Pandas GroupBy: Your Guide to Grouping Data in Python...', and 6. 'Python | Pandas DataFrame - GeeksforGeeks'.

Figure 7.2: Screenshot of the the Yippy search engine requested with the “pandas” query.

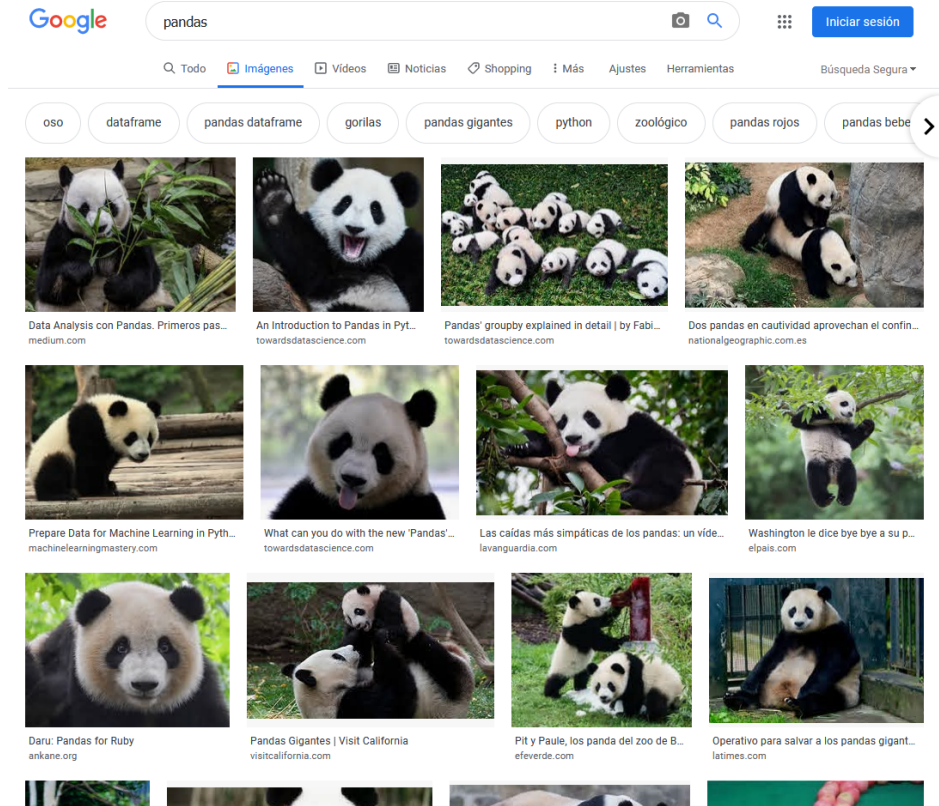


Figure 7.3: Screenshot of the the Google image search engine requested with the “pandas” query.

an evaluation of the system with users performing a sound design or a music-related task. We end the chapter with some concluding remarks.

7.2 Related work

In order to be able to organize and retrieve a large amount of poorly labeled data, automatic annotation methods have been extensively addressed in the research community (Peeters et al., 2011; Herrera-Boyer et al., 2003; Kim et al., 2010; Eronen & Klapuri, 2000; Tzanetakis & Cook, 2002; Fu et al., 2010). The main requirement of these content-based approaches is a reliable numerical feature that can represent the content. Recently, techniques using Artificial Neural Networks have been able to provide an alternative to the handcrafted features previously developed. Intermediate layers of pre-trained neural networks can serve as a higher-level representation which can be used for clustering (Jansen et al., 2017).

Clustering is a type of unsupervised classification which consists in organising similar objects in groups called clusters. Regardless of the clustering method, the content similarity measure involved is fundamental to the definition of a cluster. This similarity notion is often derived from a feature space, on which a numerical distance or similarity is calculated. When clustering audio content, the features and distance measures are often chosen carefully for a given specific task (Black & Taylor, 1997; Martins et al., 2007; Niessen et al., 2013). However in the context of large online collections, the content is very diverse, containing speech, musical or environmental sounds. This makes the choice of features and distance metric even more challenging. Among the different approaches for clustering (Fahad et al., 2014; Xu & Tian, 2015), in the context of multimedia documents, density-based algorithms such as graph-based clustering methods are particularly well designed for dealing with computational efficiency and the heterogeneous aspect of the data (Petkos et al., 2017). Moreover, in the context of sounds, graph-based algorithms based on k-Nearest Neighbors have been shown to be scalable and able to adapt to

areas of different densities (Roma Trepát et al., 2015).

In content-based methods, the media is retrieved and/or ranked using features derived from the content itself. This kind of methods makes use of signal processing and machine learning techniques to extract information from directly from the data. It has the advantage for instance to be able to retrieve poor-labeled data for the which no one provided a precise textual description. Moreover, it can provide a solution for the organisation of large amount of audio content. There are two main types of content-based methods. Some try to assign high-level concepts with the use of classifiers for instance, while others attempt to extract meaningful numerical vectors that can reflect low-level proprieties of the sounds. The first type, that we refer as *classification approaches* can be used at a pre-processing stage, in order to produce additional text metadata that can be indexed and used by traditional text search engines. The second, based on using *feature spaces*, can enable alternative methods for the browsing and the exploration of sound collections. These techniques often make use of a dimensionality reduction technique over numerical features, and project the content into a small dimensional space where similar sounds are close from each other. The user can locate a sound of interest and then explore its neighborhood. Although not many commercial tools propose this kind of alternative methods and rely only on more traditional text-based search engines, in the research community, different approaches and interfaces have been proposed. These different approaches can be considered appealing to some, by providing interaction methods that can make the process of browsing content more exciting and joyful (Schedl, 2017). For instance, spaces conveying timbral characteristics of the sounds enable quick exploration of collections (Tzanetakis & Cook, 2001). The use of visualisation tools in browsing systems have been shown to facilitate and encourage broader exploration (Wongsuphasawat et al., 2015). Some approaches rely on spectral features and organise the content into two-dimensional spaces. Rhythmic features modeled from energy frequency band fluctuations are used in (Neumayer et al., 2005a,b), combined with self-organizing map in order to provide an interactive exploration of musical spaces according to features similarity of

audio tracks. Freesound Explorer provides a visual interface for exploring sounds in a 2-dimensional space (Font & Bandiera, 2017). It allows users to perform text-based queries to Freesound, and see the results arranged in the space. It uses t-SNE as a dimensionality reduction, learned using spectral or tonality features. This way, sounds are self-organised according to some sort of timbre or harmonic similarity. Additionally, clustering identifies groups of similar sounds. Labels are assigned to each of the clusters and displayed in the space to facilitate the exploration. Sonic Browser proposes different ways to visualize and interact personal audio collections using 2-dimensional spaces, trees or graphs, and dynamic sliders filtering mechanism (Brazil & Fernstrom, 2003). In FastMap (Cano et al., 2002), the authors propose to use multi-dimensional scaling in order to discover underlying spatial structures of a set of data, from similarity information between the data. This makes this approach more general by being able to use low-level attribute, content metadata, or any underlying similarity notion. Some researchers proposed to integrate audio features with metadata into traditional search engine (Urbain et al., 2016). They combine a code-book learning approach and acoustic features in order to index content-based information into a search-engine. This enables fast content-based similarity searches, as well as visualisation with a similarity map. Clustering can be used in this feature spaces in order to automatically extract groups of similar sounds or music (Neumayer et al., 2005b). However, to our knowledge, this is the first study describing the integration of a sound clustering algorithm in a search engine. More recently, approaches for the exploration and visualisation of data have been relying on deep learning features trained on supervised tasks (Suh et al., 2017). These features seem to have the advantage of capturing more high-level characteristics of the signals, together with more low-level features. In previous chapters, the use of deep learning features such as the one obtained when training a VGGish network (Hershey et al., 2017) on large a dataset such as AudioSet (Gemmeke et al., 2017), has been proven efficient for clustering purposes, producing clusters grouping semantically-related sounds. In Chapter 5, we presented an approach that is able to learn an audio representations that can capture high and low-level charac-

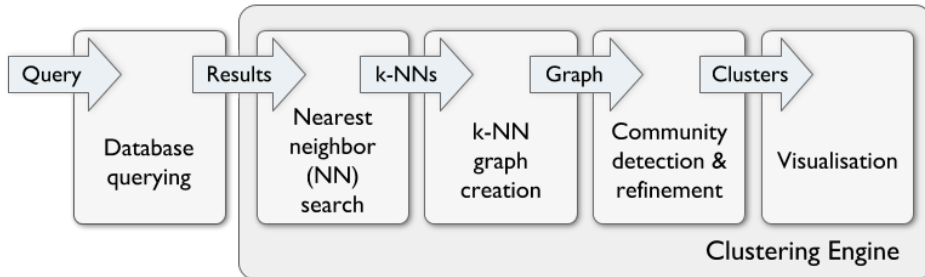


Figure 7.4: Diagram representing the steps of our clustering engine.

teristics.

7.3 Proposed approach

Our clustering engine consists of several steps illustrated in Figure 7.4. After collecting the results retrieved by the user’s query, (i) nearest neighbor searches are computed for the top ranked results, (ii) a K-Nearest Neighbors graph is created, (iii) a community detection algorithm assigns each sound to one cluster with possible extra refinements, (iv) and finally the results are displayed with a visualisation. The method is available as a service and its integration within the Freesound search engine is currently under development. The code is available at this repository: <http://omitted.for.blind.review>. We describe the audio features we will be using and comparing (Section 3.1), the graph-based clustering method (Section 3.2), a refinement strategy for discarding inconsistent clusters (Section 3.3) and the user interface of the system (Section 3.4).

7.3.1 Audio features

In this work, we compare the performance of our clustering method using three different sets of features. One set of manually selected features

Feature sets	Features
F1	spectral centroid / complexity / spread / energy energyband high / skewness / flatness db / rolloff, temporal decrease / spread / kurtosis / skewness / centroid, logattacktime, strongdecay, effective duration, zerocrossingrate, tristimulus, mfcc, dissonance
F2	lowlevel features from the Essentia Freesound Extractor ²
F3	embeddings from AudioSet pre-trained model ³

Table 7.1: The different features compared in this work.

motivated from the literature (F1) (Peeters et al., 2011; Herrera-Boyer et al., 2003; Kim et al., 2010; Eronen & Klapuri, 2000; Tzanetakis & Cook, 2002), another set contains all of the lowlevel features available from the Essentia Freesound Extractor (F2) (Bogdanov et al., 2013), and the third one uses embeddings from a neural network model trained on AudioSet (Hershey et al., 2017; Gemmeke et al., 2017) (F3). Table 7.1 details the different features.

Most of the traditional acoustic features (F1, F2) are computed on frames of approximately 50 ms. These frame-based features are summarised into a single feature vector, which ignores the temporal order. It includes minimum and maximum values, mean and variance of the direct features and of their first and second derivatives. The rest of the features, e.g. *logattacktime*, consist of a single numerical value for an entire audio clip. A dimensionality reduction is then performed using Principal Component Analysis over the entire sound collection, to reduce these concatenated statistics and values into a feature vector with 100 dimensions. The deep neural network embeddings (F3) are calculated on windows of approximately 1 second. These frame-based features are then aggregated with the mean statistic only.

7.3.2 Graph-based clustering

We use the same graph-based clustering method presented and motivated in Section 4.3.1. In the following lines we give a brief reminder and some additional information. In the graph-based clustering method, instead of directly using the features as input of a clustering method, we construct an intermediate representation of the data using a k -Nearest Neighbor Graph (Dong et al., 2011). Each vertex represents a sound, and undirected edges connect each sound to its k most similar according to the euclidean distance. Then, we use a community detection algorithm based on *modularity* optimisation for finding a partition of the graph (Blondel et al., 2008). We use the Gaia library⁴ for performing nearest neighbor searches with $\log_2(N)$ for the value of k , where N is the number of elements to cluster. Among the different reasons that motivates the usage of such a graph-based method, one corresponds to its simplicity which allows us to use some interpretable heuristics for modifying the graph or its partition like discarding clusters of low quality.

7.3.3 Discarding low quality clusters

The amount of intra-cluster and inter-cluster edges (which are related to the modularity definition (Newman & Girvan, 2004)) can be used for defining an internal quality metric which is only based on data used by the clustering algorithm. Since we use the same data representation for the quality metric and for the clustering algorithm, it is not clear if it can be used for automatically assessing the quality of a cluster in terms of compactness and distinctiveness. In this work, we are interested in investigating its use as a confidence score for quantizing the quality of an individual cluster, and possibly discard low quality clusters that should not be presented to the user in the context of Search Result Clustering. Our confidence score c of a cluster ranges from 0 to 1 and is defined as

⁴[\url{https://github.com/MTG/gaia}](https://github.com/MTG/gaia)

following:

$$c = \frac{\textit{number of intra cluster edges}}{\textit{total number of inter \& intra cluster edges}} \quad (7.1)$$

This confidence score will be higher for clusters that are more coherent, i.e., the sounds within a cluster are more similar to sounds within the same cluster than to sounds from other clusters. In the case that many of the elements of a cluster have edges to elements of other clusters, this score will be lower. This score penalises clusters that are not compact and distinct from other clusters. In this work, we investigate the use of this simple internal metric (which does not make use of any external knowledge about the data) as a confidence measure for discarding potentially irrelevant clusters.

7.3.4 Selection of cluster representative examples

In order to enhance the ability of the users to quickly have an idea of what each cluster contains, we aim at selecting representative audio examples for each cluster. The examples are selected by relying on the concept of centrality in each cluster in the graph. For each cluster, the corresponding sub-graph consisting of all the sounds belonging to the cluster, is considered. The degree of each node in its cluster is computed, which corresponds to the number of intra cluster edges. Then, we consider as representative examples, the nodes with highest degree in each cluster. In our experiment, we limit to 7 the number of selected examples.

7.3.5 User interfaces

To allow the user to interact with the clusters, we propose two different interfaces. One consists of a traditional facet filtering approach, where the user can apply filters on the result to display only sounds from one cluster. Figure 7.5 shows the modified Freesound search interface with the added clusters facets. Three labels are displayed for each cluster which correspond to the most occurring tags in the cluster. The second interface

The screenshot shows a search interface for the query "glass". At the top, there is a search bar containing "glass" and a dropdown menu set to "Automatic by relevance". Below the search bar, there is a "Cluster #1" filter. The main content area displays a list of sound results for cluster #1, each with a waveform visualization, a title, a description, a rating, and metadata. The results are:

- Glass Smash, Bottle, C.wav** (InspectorJ, April 26th, 2016, 3854 downloads, 11 comments)
- Glass Smash, Bottle, A.wav** (InspectorJ, April 26th, 2016, 2754 downloads, 3 comments)
- Glass Smash, Bottle, G.wav** (InspectorJ, April 26th, 2016, 4354 downloads, 8 comments)
- glass19.flac** (Craxtc, October 31st, 5 stars)

On the right side, there is a "clusters" section listing five clusters with their respective sound counts:

- Cluster #1 (break shatter smash, 217)
- Cluster #2 (water tone wine-glass, 280)
- Cluster #3 (break smash breaking, 62)
- Cluster #4 (broken break jar, 186)
- Cluster #5 (hit ring bowl, 242)

Below the clusters section, there is a "licenses" section listing:

- Attribution (120)
- Attribution Noncommercial (16)
- Creative Commons 0 (80)
- Sampling+ (1)

At the bottom right, there is a "tags" section with a list of related terms:

24-bit 48k 48khz bottle break breaking bris broken chest clinks crack crash crunch drop dropping falling field-recording floor foley game glass glasses gold hit percussion sample shards shatter smash verre

Figure 7.5: Page displaying the result of the query *glass* of the *cluster #1*. Clicking on a cluster facet on the right applies a cluster filter. Three labels are shown for each cluster, together with the number of sounds they contain.

consists of a 2D visualisation of the k-Nearest Neighbor Graph, where colors are used for representing clusters as shown in Figure 7.6. Sounds can be played by hovering the mouse on the nodes. Moreover, clicking on a node will highlight its neighbors in order to ease neighborhood exploration.

After performing a first user evaluation presented in Section 5, we slightly modified the prototypes. These modifications include the addition of the selected representative examples. We make them available to be listened to by adding a icon in the clustering facets which when being hovered by the mouse, plays the corresponding selected examples one after the other. In addition, in the 2D visualisation, a waveform visualisation was added when playing the sounds.

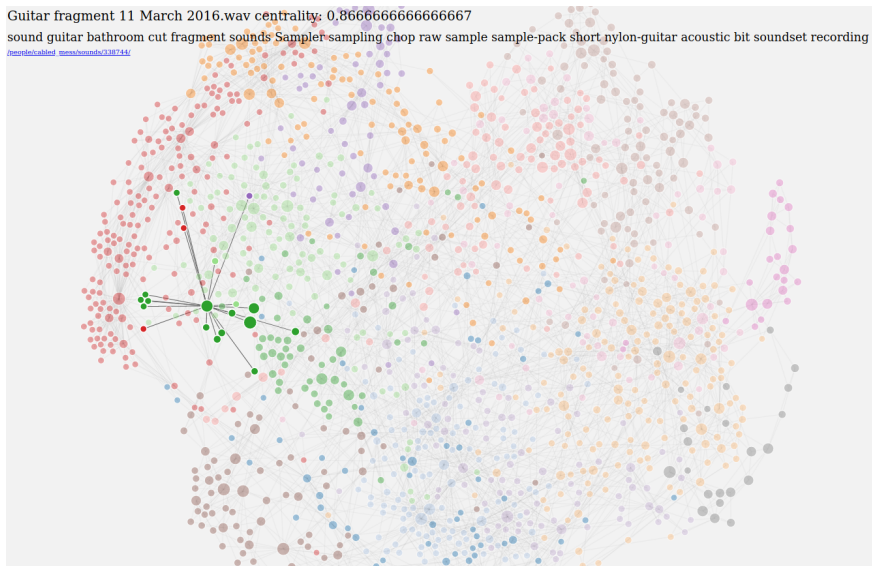


Figure 7.6: The graphical 2D visualisation of sounds retrieved with the query *guitar*. Each circle represents a sound. Placing the mouse on one will play the associated sound. Clicking on it displays some information at the top of the screen and highlights neighbor nodes.

7.4 Feature performance comparison

In this section we show some comparative performance results of our approach using the different sets of features previously described. We perform two different evaluations: one using internal validation and another using external validation. For the first one, we propose to leverage information from an existing sound sharing website to automatically evaluate the clustering performances at scale. We then perform a standard evaluation which uses ground truth labels taken from manually annotated datasets. One of the goals to perform these two evaluations is to validate that our first evaluation that does not require known ground truth labels is adequate for comparing the performance of different clustering methods.

7.4.1 Internal validation

We consider the Freesound database as a use case and we perform clustering on the search results of popular queries submitted by real users of the platform. We focus only on sounds with duration from 0 to 10 seconds. For a quantitative evaluation, we make use of the sounds' metadata that is provided by the creator of the content in the Freesound platform.

Evaluating a clustering automatically is a complicated task, and there are different types of metrics that can be used (Manning et al., 2010). Some of them are referred as internal metrics, and they are used when no ground truth label is known (Liu et al., 2010). The Calinski-Harabasz Index (CHI) (Caliński & Harabasz, 1974) evaluates the cluster validity based on the average between- and within-cluster sum of squares as shown in this equation:

$$CH(k) = [B(k)/W(k)][(n - k)/(k - 1)] \quad (7.2)$$

Where n corresponds to the number of data points, k to the number of clusters, $W(k)$ to the within cluster variation and $B(k)$ to the between cluster variation.

Instead of calculating this metric using the audio features used for clustering, we make use of the user-provided tags associated with the audio content as an external information. This allows us to evaluate the

overall quality of a clustering, from a semantic perspective. From the tags associated to the content, we derive a feature using a Vector Space Model representation (Salton, 1989). This feature is a high-dimensional sparse vector where a value of 1 in one dimension refers to the presence of a specific tag. We only consider the 5000 tags that occur the most in the overall Freesound collection. We then reduce the size of this vector to 100, by applying Latent Semantic Analysis, which can capture synonymy relations (Deerwester et al., 1990). Due to the nature of tags, the validity metric we use is not always accurate. In order to mitigate this problem, we average this metric on clusterings performed on the results of the 1000 most popular queries in Freesound. In total, approximately 80k different sounds were used in this evaluation. Figure 7.7 represents the evaluation pipeline. The statistics are presented in Table 7.2. A discussion of the results is given in Section 7.4.3.

7.4.2 External validation

As an additional evaluation, we also make use of an external validation metrics, which relies on known ground truth labels. We exploit data gathered within Freesound Annotator (Fonseca et al., 2017b) to construct 44 datasets comprising in total around 30k sounds and 215 different labels. Labels are drawn from the AudioSet Ontology (Gemmeke et al., 2017), which consists of a hierarchical taxonomy of 635 sound-related categories. In our experiment, a dataset consists of one node in the taxonomy, and its labels are its direct children. This creates datasets of different sizes and with different levels of specificity. For instance, one broad dataset corresponds to natural sounds, containing the *water*, *wind*, *thunderstorm* and *fire* classes. A more specific one contains only water sounds, with the *rain*, *stream*, *steam*, *waterfall*, *gurgling*, and *ocean* classes. All the datasets contain sounds with a duration lower than 10 seconds and most of them contain only one salient source, which mitigates the inconvenience of using a statistical aggregation over the frame-based features. Among the popular metrics used for comparing dataset partitions, the literature suggests that Adjusted Mutual Information (AMI)

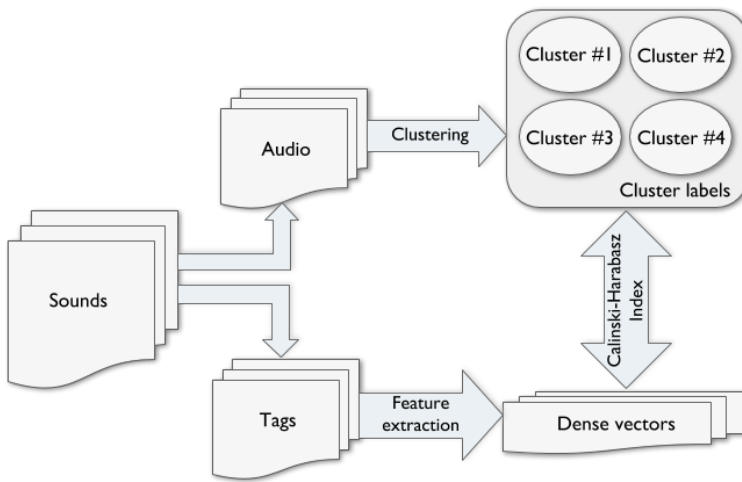


Figure 7.7: Diagram representing the steps of our internal evaluation making use of user-provided tags. The Calinski-Harabasz Index is calculated between the labels corresponding to the obtained clusters and the features derived from the sound tags. This evaluation is performed on the results of the 1000 most popular queries performed by Freesound users.

score (Vinh et al., 2010) is suited when the reference clustering (ground truth) is unbalanced and there exist small clusters (Romano et al., 2016). This corresponds often to what we find in collaborative collections where the content is inconsistently distributed in terms of type and nature. For evaluating the different sets of features with the different features, we perform clustering on the datasets and measure the similarity between the real partition (given by the ground truth labels) and the one given by the clustering methods by computing the Mutual Information score adjusted for chance (AMI), calculated as following:

$$AMI(U, V) = \frac{MI(U, V) - E\{MI(U, V)\}}{\max\{H(U), H(V)\} - E\{MI(U, V)\}} \quad (7.3)$$

$$MI(U, V) = \sum_i \sum_j P(i, j) \log\left(\frac{P(i, j)}{P(i)P(j)}\right) \quad (7.4)$$

Where U and V are the two partitions to compare, $H(U)$ and $H(V)$ their associated entropy. $P(i)$ and $P(j)$ the probabilities that a point belongs to cluster U_i or V_j respectively, $P(i, j)$ the probability that a point belongs to both cluster U_i and V_j from U and V respectively, and $E\{MI(U, V)\}$ corresponds to the expected mutual information between two random clusterings. The Mutual Information metric (MI) quantifies the information shared by the two partitions and therefore can be used as a clustering similarity measure. When adjusted for chance, the metric takes a value of 1 for two identical partitions and the value of 0 for two randomly dissimilar partitions. Table 7.3 shows some statistics of this score for the different audio features.

7.4.3 Results

In both evaluations, AudioSet embeddings lead to the best clustering performance. This shows that novel deep learning approaches can produce semantically meaningful features that outperform traditional handcrafted features for the unsupervised classification of sounds, which is in line with our findings from Chapter 4. There is not any meaningful difference

Features	CHI			
	no pruning		pruning	
	mean	std	mean	std
F1	3.36	5.87	3.88	7.25
F2	3.44	6.37	3.86	7.07
F3	4.29	6.82	5.29	11.06

Table 7.2: Clustering validity score (Calinski-Harabasz Index) using the different feature sets. Mean and standard deviation is calculated on the performance of the clustering of the results from the top 1000 most popular queries in Freesound. The pruning column corresponds to the validity score when discarding the cluster with the lowest confidence score defined in Section 3.3.

when applying manual feature selection over handcrafted features (F1) motivated by results taken from the literature compared to using a large set of low-level features (F2).

Our approach for discarding low quality clusters using the confidence measure described in Section 3.3 shows little but consistent improvement with all the features in both experiments.

Another conclusion is that our proposed internal validation which makes use of accompanying tags provided by the users of the platform can give similar results as an external validation using ground-truth labels. This provides a valuable framework for evaluating clustering algorithms in existing multimedia collections at scale, without needing to manually annotate a large amount of data.

7.5 User evaluation

In this section, we present our user-centered design process on the development of an interface for browsing sounds from large databases using the proposed clustering engine. In this experiment, we use the AudioSet

Features	AMI			
	no pruning		pruning	
	mean	std	mean	std
F1	0.16	0.08	0.18	0.10
F2	0.15	0.09	0.16	0.12
F3	0.20	0.10	0.21	0.11

Table 7.3: Average performance (AMI) across the different dataset with the different features. An AMI close to 0 corresponds to a random partition while perfect matches gives 1 AMI. The pruning column corresponds to the performance when discarding the cluster with the lowest confidence score defined in Section 7.3.3.

features (F3), which achieved the best performance in our comparative performance evaluation. We use our interface prototypes as *technology probes* to observe their use in a real context, to evaluate their functionalities and to inspire new ideas (Hutchinson et al., 2003).

7.5.1 Methodology

We performed 2 experiments in an iterative fashion. We first evaluated our system using the first interface prototype described in Section 3.4. Then, we included the modifications previously described and repeated a second user experiment. Both experiments were performed with 4 different users that are experienced with audio and sound design tasks, which is sufficient for detecting a large amount of usability problems (Nielsen, 2000). All the participants were presented with the two searching tools (clustering facets filtering and 2D visualisation) presented in Section 3.4. The first task consisted in gathering all the audio content needed in order to build the soundtrack of a short video, available at: <https://vimeo.com/333837958>. This video was chosen because it was short but presenting a lot of variety of elements to sonify. The original sound was removed from the video. The second task was musically oriented. The users had

to listen to short music loops, and asked to gather all the content needed to be able to rebuild them in their own way. Loops were selected from commercial music, including a wide variety of genres and instruments sounds. Both tasks can be considered as more searching tasks rather than exploratory.

Some guidelines were shown to the participants, together with verbal explanations given by the examiner who was present during the entire experiment. At the end of the task, they were provided with a questionnaire containing some usability and engagement questions. Finally, semi-structured interviews were carried out, including open-ended questions as well as specific questions related to observed behaviors during the performance of the task. This enables discussion using thematic analysis in order to identify emerging themes from participants' answers.

7.5.2 Results and discussion

All the participants started by watching the video and noting down the concepts they would then look for within the audio collections. Then, they started to use the search engine to look for the identified concepts needed for sonifying the video or the instruments used to reproduce the music loops. After entering a query, users often had a quick look at the top results. They explained that it allowed them to figure out if the content retrieved was the one they were expecting. They then had the choice to either reformulate their query, or explore the retrieved results. They found particularly useful the labels associated with each clusters in order to identify what kind of sounds were present in the results and what type of content each cluster contained. They were either applying a cluster filtering to then browse the results in the retrieved sounds, or they would use terms from the cluster labels to reformulate their query.

Some participants complained that the cluster labels were sometimes inappropriate, because they contained too broad concepts, or they were very similar for different clusters. In these cases, it was hard to understand what type of content each cluster contained. Nevertheless, the 2D visualisation was then particularly helpful. The participants were often

listening to many sounds in a short amount of time thanks to the the fact that they only needed to hover the mouse on different nodes to start hearing some sounds. Whereas in the flat ranked list, they would have to manually trigger the players by clicking many times. In the 2D visualisation, their strategy was to first listen to a few dispersed sounds to quickly get an idea of how the sounds were organised in the space and what type of content was present in each cluster. Then, they would start exploring specific regions of interest, until they satisfy their need by retrieving one or several relevant sounds. In addition, the users were often searching in a sound's neighborhood. They explained that they wanted to find some slightly different variations of a relevant sound they already located. However, understanding what the dimensions were capturing in the space was difficult. One participant reported that the graph representation of the sound results was failing for instance to reflect timbral characteristics of the sounds in a clear way. Moreover, even if the graph was presenting a clear structure, it was not easy to understand to what it was corresponding to and to locate all the relevant content. As a solution, a participant wanted to be able to select any retrieved sound from the ranked list and locate it in the 2D visualisation. This way, he explained that he could easily switch from one interface to the other, allowing him to efficiently combine the two interaction approaches. Moreover, several users complained that in the 2D visualisation, no labels were presented, and it was therefore hard to associate the clusters from one interface to the other. As a solution, the idea of adding label information for each cluster in the 2D visualisation was discussed.

The clustering engine was not always beneficial when the participants were using precise queries containing multiple words. In the context of sound design for example, they explained that they often know exactly what they need. And therefore they usually formulate a precise text query retrieving very specific content. However, one drawback is that sounds that would not present the query terms as metadata would not be retrieved. A solution to deal with bad recall performance of the system would be to use the audio-based representation to expand the retrieved results with sounds that are similar to the one retrieved. In its current state, the proto-

type only applies clustering on the retrieved results, but does not include sounds that could have been relevant, but were missed to be retrieved by the text-based search engine. Using the audio-based features for expanding the retrieved results would be interesting to study for queries that retrieved very few results.

Some participants of the first experiment criticised the fact that the 2D visualisation did not provide any representation of the waveform or any time-related information regarding the audio clips. This made it hard to explore some results, as many of the users actively use waveforms in order to identify for instance if a foreground sound would appear at some time in the audio clip. For that reason, in the 2D visualisation, many participants were skipping some audio clips because the main acoustic event was not starting at the beginning of the clip. Moreover, some participants said that they often use the waveform representation to assess some characteristics of the sounds, such as its dynamics, or the level of background noise present in the audio clip. Displaying the waveform with a time progression cursor of the current sound being played is therefore a key feature that would make the 2D visualisation more useful. However, the waveform visualisation that was added in the 2D visualisation was not very useful for the participant of the musically related task. Contrary to the first experiment, where participants were searching for environmental sounds, in the second experiment, most of the retrieved content was consisting of instrument samples or loops, that did not have multiple sources that would occurring later within the clip. Often, musical content is well segmented because users in Freesound upload loops or samples that can be sometimes directly used without further edition processing.

In its current state, the clustering algorithm was able to discover distinctive and coherent groups in search results for many given queries. However, in some cases, the quality of the clustering is still low, which made some users wasting time exploring bad clusters. Moreover, they explained that spending time exploring non-relevant clusterings could make them lose trust in the system, and therefore make them not use it again. It was discussed in the interview the idea of reporting an estimate of the quality of the clustering, so that the user would be aware of its poor result

and would be more confident while using the system. Using a confidence measure such as the one proposed in Section 3.3 could be a solution, or using directly the modularity of the graph partition. Our evaluation of the strategy for discarding low quality clusters in Section 7.4 indicates that such measures can reflect, to some extent, the quality of the clustering.

For the second experiment, we added the possibility for the users to listen to sound examples extracted from the clusters. The users were asked if in their opinion the audio examples seemed to correspond to the cluster labels. Some users answered that the examples were not always corresponding to the labels. Moreover, they argued that the examples were distracting them sometimes, making them loose time. However, they said that the examples were more useful to identify if a cluster could be relevant, which allowed for instance for one participant to know if it was worth going to explore the 2D visualisation. Finally, one participant explained that he was sometimes getting frustrated because he found one of the examples relevant for his task, but could not find a way to easily have access to it for downloading it.

During the first experiment, it was commented by one user that the way the clustering facets were applied was maybe not appropriate. For the two experiments, the clustering was performed on the results obtained after any filtering process. This means that if a user applies a traditional facet filter, such as filtering by tag or samplerate, the system would make the clustering facet disappear and compute a new clustering on the new returned results. While the main idea of this behavior was to be able provide more possibilities of use of the clustering engine, by enabling the user to get a huge amount of different clustering results by simply combining filters, it seemed that this behavior could make the system a bit cumbersome and inconsistent. Indeed, a user that is used to the functioning of facets, would expect that clustering facets would work the same way as the classical facets. This means that combining facets is possible, and that applying them together applies several filter over the facets' content. This idea of having the same behavior for the clustering facet as the classical ones was then discussed more in depth with the participants of the second experiment, since it was something that was overlooked dur-

ing the first. Most of the users agreed that what would make more sense to users would be to have the clustering facet working the same way as the classical facet. Some proposed that the ability to cluster results after applying filtering with traditional facets could be enabled by adding a button for requesting the re-computation of the clusters.

7.6 Conclusion

In this chapter we present a Search Result Clustering approach for enabling users to browse large online sound collections. To our knowledge, it is the first time that such an approach is applied in the context of sounds retrieval. We perform audio clustering using a graph-based approach which is relatively fast to compute and has the advantage of not requiring to specify the number of clusters in advance. We carried two evaluations for comparing the performance of different features. The first one uses data of an online collection for accessing the performances at scale, whereas the second makes use of ground-truth labels from a reduced-size manually annotated collection. Performing two evaluations enables the performance comparison to get more credibility. Moreover, the results suggest that using tag metadata associated with the audio files can enable to perform an evaluation that can give comparable results as using manually assigned ground truth labels. This type of automatic evaluation can facilitate the development of Search Result Clustering approaches in other contexts, where an automatic evaluation at scale can be obtained without the need of manually defining ground truth labels or partitions. Results correspond to what has been observed in Chapter 4, where we compared the performance of different types of features. Embeddings obtained by training neural networks on a supervised classification task with large amount of data can be used as a feature that increases the performance of the clustering compared to more traditional handcrafted features.

We also investigated the use of methods for discarding low quality clusters based on the graph structure and its partitions. An heuristic involving the ratio of intra-cluster edges and the total number of edges in

each cluster was able to improve the performance of the clustering in the two automatic evaluations, which shows that the proposed approach is able to discard clusters of low quality. Moreover, it suggests that structure of the graph and its partition can be related to the quality of the partition, which can for instance enable methods for automatically assessing the quality of a clustering partition without relying on any extra information.

Finally, the system was integrated into the Freesound search page, where the users can interact with the clusters using two different interfaces. One consists on applying filters on the retrieved results, and the other involves a 2D visualisation of a graph representation of the sounds. We evaluate the system in the context of real-world sound design and music making tasks. Our results suggest that Search Result Clustering can assist the browsing of large sound collections. Interesting feedback was gathered which will guide future development of the clustering engine and its integration within the Freesound platform. Furthermore, the methodology followed in this work provides a valuable framework for developing and evaluating clustering engines in the broad area of multimedia content retrieval. As a result to these experiments, we implemented a new version of the interfaces. Since users showed interest in navigating from the ranked list interface to the 2D visualisation, we first decided to embed the graph visualisation in a modal on the same page as the search engine is on. This would allow moving from one visualisation to the other in a smooth way. Then, we added a functionality in order to directly locate a cluster in the graph visualisation from the clustering facets. Clicking on a cluster facet *locate* button, opens the graph visualisation and highlight the desired cluster. As it was discussed during the interviews with some users, we added some labels in the 2D visualisation in order to facilitate the identification of what each cluster corresponds to. The current state of the interfaces are presented in Figure 7.8 and Figure 7.9.

The screenshot shows the freesound.org search results for the query 'guitar'. The page layout includes a header with navigation links (Register, Log In, Upload Sounds), a search bar, and a sidebar with filters (Sounds, Forums, People, Help). The main content area displays search results for 'guitar', showing audio waveforms, sample names (e.g., BD_1.wav, BD_4.wav, BD_3.wav, squeak4.wav, rock band (the game cont...), xrigamr), and descriptions. A 'clusters' section on the right lists 7 clusters of related terms, and a 'tags' section lists various audio-related tags.

clusters slow

- Cluster #1 (36)
- echomachine mpc bass
- Cluster #2 (25)
- echomachine mpc distorted
- Cluster #3 (16)
- distortion distorted 24bit
- Cluster #4 (1)
- mpc echomachine
- Cluster #5 (12)
- mpc echomachine piano
- Cluster #6 (8)
- synth string loops
- Cluster #7 (9)
- electric squeak string

tags

analog atmospheric bass beat
 chomachines chord data emf creak drum
 echomachine ep fi fruity groove
 guitar hip hop loop loops
 manipulated mpc mtc500-m002-s14
 music pitch squeak studio techno trance

Figure 7.8: Results from the *guitar* query in the search engine. Clustering facets behave like the traditional one, enabling users to combine different types of facets to further narrow down the search results.

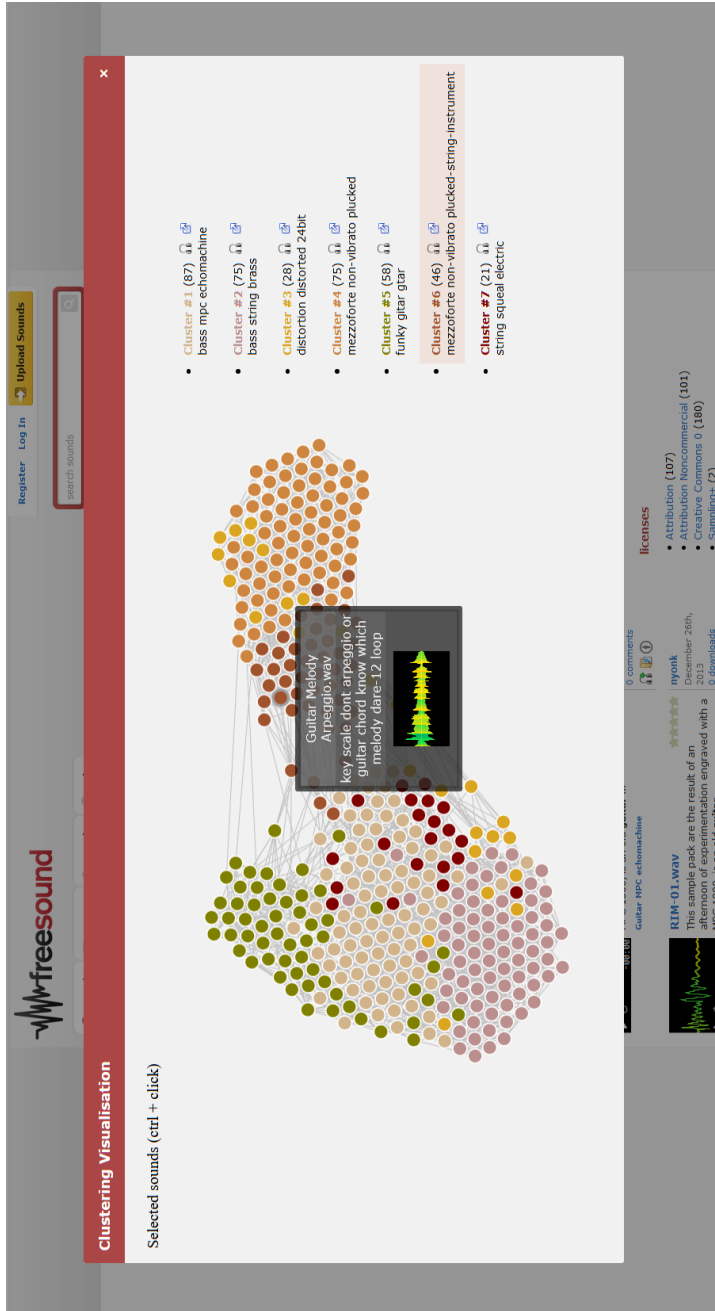


Figure 7.9: The 2D visualisation displaying the kNN-graph for the *guitar* query.

Chapter 8

Conclusions

In this chapter, we start by providing a summary of the main contributions presented in this thesis. We then discuss its findings and highlight perspectives on new methods and research opportunities identified as promising for the organization and retrieval of audio content. The discussion is structured following the list of research questions:

- (i) How can we make best use of sound collaborative collections in order to build high-quality datasets for supporting advances in machine learning?
- (ii) To what extent and in what way can collaborative collections be directly used to learn audio representations that are useful for classification tasks?
- (iii) How do deep learning features perform for unsupervised classification with a wide variety of sounds?
- (iv) How feasible and valuable is Search Results Clustering for retrieving content from collaborative sound collections?

8.1 Summary of contributions

This thesis contributes to the advancement of the state of the art from different perspectives in order to improve the retrieval of sounds in large collaborative collections. The main contributions of this thesis can be summarized as follows:

- It identifies ideal characteristics of high-quality datasets that can contribute to advances in machine listening, and proposes a methodology for building such datasets using data from online sound collections.
- It contributes to the creation a number of datasets that have a great impact in the research community, by allowing researchers to develop and evaluate auto-tagging models. These models are of great

importance for collaborative sound collections, which suffer from being non-uniformly annotated. These models can generate new annotations that can be indexed in order to improve the retrieval system.

- It proposes novel manual annotation tools that successfully enhance the ability of users to understand and adopt large taxonomies. This contributes to obtain better datasets, as well as facilitating the improvement audio-related taxonomies, by potentially democratizing their usage, allowing them to be used by a wider audience and to be kept under constant scrutiny.
- It proposes successful methods for learning audio representations by taking advantage of sounds and their metadata. Given the enormous amount of online data and its constant growth, its complete annotation remains a big challenge. The developed approaches can take advantage of the abundance of data, without needing to build high-quality datasets. The learned representations can serve as a base for many classification tasks, including auto-tagging and clustering, which can eventually improve the retrieval of sounds.
- It provides a number of methodologies to compare the performance of different features and methods for the unsupervised classification of sounds. This allows us identifying what features and methods are appropriate for clustering a wide variety of sounds.
- It provides a Search Results Clustering method applied to sounds and integrated into Freesound, which has been shown to improve the retrieval of sounds. The system is planned to be deployed in the platform and used daily by thousands of users.

The research carried out in this thesis has been published in the form of several papers in top international conferences. The outcomes of Chapter 2 have been published in a conference paper (Fonseca et al., 2017a) and in a pre-print journal paper (Fonseca et al., 2020a). The investigation presented in Chapter 3 has been published in a conference paper (Favory

et al., 2018). Furthermore, the outcome of the research carried out in Chapter 5 has been published in a conference (Favory et al., 2020a). The outcomes of Chapter 6 have been submitted to a conference (Favory et al., 2020b). Finally, the work presented in Chapter 7 has been published in a conference (Favory et al., 2020c). The full list of the author’s publications is provided in the Appendix.

8.2 Building high-quality datasets

The creation of datasets by leveraging data from the Freesound database was one of the contributions of this thesis with the greatest impact in the research community. Chapter 2 highlights the ideal characteristics of such datasets, meeting the requirements of a wide range of applications related to machine listening and music information retrieval. In addition, we focus on investigating the problem of manual annotation of audio content. Due to the growing amount of available data, exhaustive annotation requires many human contributors, which is, for large datasets, commonly addressed through crowdsourcing (Gemmeke et al., 2017; Law et al., 2009; Deng et al., 2009; Drossos et al., 2020). However, instead of relying on external crowdsourcing platforms, such as Amazon Mechanical Turk or Figure Eight (formerly Crowdfunder), we decided to implement and maintain our own platform. Reasons for that include the fact that the requirements of the annotation process we designed could not be met by existing platforms. For example, we acquired high-quality annotations, using a very specialized task that requires experts and a custom interface. In addition, the use of external crowdsourcing platforms is not sustainable as a long-term annotation solution, which is what is aimed at in the case of Freesound. To give an idea, we have been collecting contributions from more than 500 users for over 3 years before releasing the FSD50K dataset, gathering over 300k contributions. We were able to design our own instructions, test cases, prioritization scheme to acquire data, and thus build datasets faster. Notably, were able to design experiments to investigate novel annotations tools.

The latter was the focus of Chapter 3, where we explored the feasibility of more advanced annotation tasks in a collaborative context. Producing consistent and exhaustive annotations for generic datasets calls for tools for handling large numbers of categories. In particular, we decided to concentrate on facilitating the exploration and use of predefined sets of categories taken from large taxonomies.

This research has allowed us to identify several opportunities for improving data collection in sharing platforms. As an alternative to annotating the content outside the original sharing platform, the tools developed during the course of our research can help the annotation early-on at upload time, if included directly in the platform. For example, users could annotate their content using a pre-defined set of categories. At upload time, the user could be prompted with a set of categories recommended based on user-specific tags. The recommended categories could then be refined or made more precise using a tool like the *Refinement Annotator*, which produced consistent annotations between distinct users, as shown in Chapter 3.

Furthermore the *Manual Annotator* provides an intuitive interface to search and retrieve categories from large taxonomies, and could be useful in the exploration of sounds. An intuitive interface for category exploration democratizes the use of taxonomies, allowing them to be used by a wider audience and to be kept under constant scrutiny. This would facilitate the maintenance of taxonomies, which need to constantly adapt to a changing landscape of sounds. For example, if a new instrument is invented or an unknown animal sound is recorded, taxonomies should be maintained in a way that quickly identify and integrate these new elements.

Promising directions of research towards improving the automatic organization of audio content, which were not explored in this thesis, include *active learning* (Kholghi et al., 2018), *lifelong learning* (Parisi et al., 2019), and *never-ending learning* (Mitchell et al., 2018). Rather than annotating content comprehensively, active learning concentrates on selecting those cases whose annotation is more informative and has a greater impact on classification performance. This could potentially lead

to significant reductions in the number of sounds that need to be annotated. Similarly, lifelong and never-ending approaches could be used to construct a content-based auto-tagging system that learns continuously throughout time. This system could be used to recommend tags, in a similar way it is done in Freesound using a purely tag-based recommendation system (Font et al., 2013b). If a user accepts or discards some of the recommended tags, the model updates itself to provide better predictions and recommendations in the future. More sophisticated approaches could also take advantage of other behaviors of the users of online platforms. Search, listen and download history can provide relevance feedback, which could be used to improve the search experience, as proposed by Qi et al. (2020). Social features such as ratings, comments, number of downloads provide rich information and could be also taken into account for other aims, such as automatic quality assessment.

8.3 Audio representation learning

Deep neural networks are able to learn valuable audio features when trained with a large amounts of annotated data. As previously discussed, obtaining high-quality datasets is difficult and time consuming. As an alternative to supervised methods, unsupervised and self-supervised approaches can make use of larger amounts of data, by using content that does not need to be annotated. There is a mid point between highly-curated datasets and non-annotated datasets which is the data arising from collaborative collections. This data and its accompanying metadata are obtained directly from the platforms, without post-processing. Having models that take advantage of such content without the need for resource-intensive curation would potentially increase the amount of data available. In Chapter 5 and 6, we introduced methods for learning audio features, by aligning the learned latent representations of audio and associated tags taken from Freesound. The popularity of contrastive learning has grown substantially in the last years (Le-Khac et al., 2020). Triplet loss is one of the most popular metric learning approaches to learning content fea-

tures (Weinberger & Saul, 2009). However, sampling informative triplets that are crucial to the learning process requires significant effort (Won et al., 2020). Moreover, this approach involves triplets of data points in the formulation of the loss, possibly missing information about relationships between all members within a mini-batch. The recently introduced *infoNCE* (Oord et al., 2018) or *NT-Xent* (Chen et al., 2020) losses overcome these difficulties by involving all the data points within a mini-batch when training. Employing these loss functions in a self-supervised way has led to powerful image and audio representations learning, without the need for annotated data (Chen et al., 2020; Fonseca et al., 2020b). We investigated the use of this type of losses in an heterogeneous setting, where we make use of two associated modalities: audio and tags. We demonstrate that it is possible to leverage the accompanying tags in order to learn high-level features in a relatively straightforward way. Moreover, in order to learn a feature that reflects both semantic and low-level acoustic characteristics, we combine contrastive learning with a reconstruction objective by employing an autoencoder architecture.

Our results are promising, sometimes in par with the state-of-the-art deep features for audio classification tasks. Furthermore, the embeddings obtained when using the reconstruction objective are more correlated with acoustic features, making the learned embedding suitable for a larger number of tasks. As an example, using the reconstruction objective led to improvements in musical instrument classification.

Regarding the tag-based model, the multi-hot encoding tag representation given as input hinders generalization to terms unseen during training and is susceptible to the curse of dimensionality, as the size of the vocabulary increases. We therefore proposed the use of pre-trained word representations combined with a self-attention mechanism in order to learn aggregated and contextualized tag representations (Chapter 6). The resulting audio embeddings reached performances comparable to the previous approach, with potential for improved semantic generalization.

The idea of generalization is however not explored in this manuscript and remains an open direction of research. For example, we could evaluate the performance of the learned audio embeddings on different tasks,

e.g. zero-shot learning (Choi et al., 2019; Schonfeld et al., 2019). As another promising direction of research, contrastive learning with heterogeneous encoders can be augmented by including additional information, e.g. text descriptions instead of tags. This would make the approach more suitable when using data from various other online platforms, such as YouTube, which do not employ a tag-based annotation system. In this case, relying on more sophisticated text-based models, such as transformers (et al., 2017), can be beneficial, since extracting semantic information from text is likely to be much harder than doing so from tags.

The idea of combining different sources of data could be extended to the use of distinct modalities. As such, we believe that associating different multimedia modalities, such as video, could bring about important contributions in representation learning, potentially serving as a base for many applications. For instance, recent approaches attempt to learn representations by combining audio and image data (Arandjelovic & Zisserman, 2017; Cramer et al., 2019; Surís et al., 2018), while other approaches combine video and text (Aytar et al., 2017; Sun et al., 2019). Since our approach learns through aligning latent representations, it has the potential to learn from visual, audio and text correspondences. Such a multi-modal approach could have a significant impact in different fields, making this an interesting line of future research.

8.4 Feature performance for clustering

In Chapter 4, we aimed at identifying which type of audio features is the most suited for clustering sounds from large and diverse online sound collections. To this end, we compare the performance of five sets of audio features, using two different clustering algorithms. In addition, in Chapter 7, we conducted an evaluation of three sets of features at scale, in the context of Search Results Clustering.

Features targeted for classifying the wide variety of sounds available in online collections such as Freesound are not so common, i.e. features and systems are usually designed with a specific purpose in mind. More-

over, the performance of features for unsupervised sound classification are rarely reported in the literature. Even though deep neural network embeddings are believed to perform well in this context, we wanted to clarify their merit when clustering diverse types of sounds. Our experiments indicate that embeddings trained with large datasets annotated based on large taxonomies, such as AudioSet, lead to superior clustering performances more often than the competing methods. Self-supervised techniques employing, for instance audio and visual associations, also provided good features (OpenL3). However, the latter did not appear to perform as well as features learned from large annotated datasets. In addition, our results suggest that deep learning features have the potential to yield better results if learned based on and applied to data from more specialized domains. As an example, features learned exclusively with environmental sounds tend to produce better clustering performance in this specific scope. This can be important in the context of Search Results Clustering, where features learned with different datasets could be selected according to the query entered by the user (e.g., music related or environmental sounds).

An essential step in proposing features for clustering is to have a simple and reliable way of evaluating them. Therefore, in Chapter 7, we propose the use of tag metadata from Freesound to evaluate clustering at scale, avoiding the need for external ground truth information. This approach was able to mirror the results obtained with the external validation based on ground truth data, although a more extensive comparison would be informative. This evaluation approach allows researchers from other fields to assess their content-based features and clustering performances at scale, using multimedia data with the associated tags and descriptions.

8.5 Search Results Clustering

In this thesis we investigated the use of Search Results Clustering (SRC) for helping users to browse large online sound collections. To our knowledge, it is the first time that such an approach is applied in such context. We perform audio clustering using a graph-based approach that is rela-

tively fast to compute and does not impose the use of a predefined number of clusters. The results indicate that SRC can significantly improve browsing of large sound collections. In addition to the clustering algorithm, we complement the system with several user interfaces, which allow the users to interact with the clusters using (i) *facets* to filter the results with content from a specific cluster, and (ii) through an exploratory interface that displays a nearest-neighbors graph in an intuitive 2-dimensional space. The system was evaluated in the context of real-world sound design and music making tasks.

On the one hand, our experiments indicate that strategies to improve sound retrieval can significantly improve the precision and relevance of the search results. On the other hand, the users' knowledge and understanding of the content of the collection. The former refers to the fact that the proposed strategies narrow down the number of results, allowing the users to easily retrieve content that would otherwise be missed because it would not appear as a top result. The latter refers to the use of clustering facets and their labels, allowing the users to adjust and expand their vocabulary towards that of the collection, and so reformulate their queries to better express their needs. In addition, the 2D visualisation tool provides a fast means of exploring large sets of results, while capturing the structure and relationships between the obtained clusters.

According to our internal log history, Freesound serves more than 200k queries a day, corresponding to more than 3 queries per second. Therefore, integrating SRC into Freesound has the potential to significantly impact the creative needs of a large number of users. However, the integration of SRC at such a scale raises important computational challenges. As a consequence, the successful deployment of SRC calls for significant engineering efforts into making its execution scalable.

Clustering also has the potential to speed up the manual annotation process by, for instance, allowing the user to annotate only a single or a few sounds per cluster. The system could then take advantage of this information by extrapolating those annotations to other sounds within the cluster. On one hand, we showed that clustering methods improve sound retrieval. On the other hand, we demonstrated how approaches that fa-

Facilitate manual annotation promote the creation of high-quality datasets. Combining the two strategies can potentially lead to novel active learning methods that can save annotation efforts (Shuyang et al., 2017).

Bibliography

- Abu-El-Haija, S., Kothari, N., Lee, J., Natsev, P., Toderici, G., Varadarajan, B., & Vijayanarasimhan, S. (2016). Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*.
- Alelyani, S., Tang, J., & Liu, H. (2013). Feature selection for clustering: A review. *Data Clustering: Algorithms and Applications*, 29, 110–121.
- Alonso-Jiménez, P., Bogdanov, D., Pons, J., & Serra, X. (2020). Tensorflow audio models in essentia. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 266–270.
- Amiriparian, S., Freitag, M., Cummins, N., & Schuller, B. (2017). Sequence to sequence autoencoders for unsupervised representation learning from audio. In *Proc. of the DCASE 2017 Workshop*.
- Arandjelovic, R. & Zisserman, A. (2017). Look, listen and learn. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 609–617.
- Arbelaitz, O., Gurrutxaga, I., Muguerza, J., Pérez, J. M., & Perona, I. (2013). An extensive comparative study of cluster validity indices. *Pattern Recognition*, 46(1), 243–256.
- Aumüller, M., Bernhardsson, E., & Faithfull, A. (2019). Ann-

- benchmarks: A benchmarking tool for approximate nearest neighbor algorithms. *Information Systems*.
- Aytar, Y., Vondrick, C., & Torralba, A. (2016). Soundnet: Learning sound representations from unlabeled video. In *Advances in neural information processing systems*, pp. 892–900.
- Aytar, Y., Vondrick, C., & Torralba, A. (2017). See, hear, and read: Deep aligned representations. *arXiv preprint arXiv:1706.00932*.
- Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization.
- Balzer, M. & Deussen, O. (2005). Voronoi treemaps. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, pp. 49–56. IEEE.
- Barchiesi, D., Giannoulis, D., Stowell, D., & Plumbley, M. D. (2015). Acoustic scene classification: Classifying environments from the sounds they produce. *IEEE Signal Processing Magazine*, 32(3), 16–34.
- Bello, J. P., Silva, C., Nov, O., DuBois, R. L., Arora, A., Salamon, J., Mydlarz, C., & Doraiswamy, H. (2019). Sonyc: A system for monitoring, analyzing, and mitigating urban noise pollution. *Communications of the ACM*, 62(2), 68–77.
- Ben-Haim, N., Babenko, B., & Belongie, S. (2006). Improving web-based image search via content based clustering. In *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, pp. 106–106. IEEE.
- Berkhin, P. (2006). A survey of clustering data mining techniques. In *Grouping multidimensional data*, pp. 25–71. Springer.
- Bertin-Mahieux, T., Ellis, D. P., Whitman, B., & Lamere, P. (2011). The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*.

- Black, A. W. & Taylor, P. A. (1997). Automatically clustering similar units for unit selection in speech synthesis.
- Blondel, V. D., Guillaume, J.-L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10), P10008.
- Bogdanov, D., Wack, N., Gómez Gutiérrez, E., Gulati, S., Herrera Boyer, P., Mayor, O., Roma Trepat, G., Salamon, J., Zapata González, J. R., & Serra, X. (2013). Essentia: An audio analysis library for music information retrieval. In *Britto A, Gouyon F, Dixon S, editors. 14th Conference of the International Society for Music Information Retrieval (ISMIR); 2013 Nov 4-8; Curitiba, Brazil.[place unknown]: ISMIR; 2013. p. 493-8. International Society for Music Information Retrieval (ISMIR).*
- Brazil, E. & Fernstrom, M. (2003). Audio information browsing with the sonic browser. In *Proceedings International Conference on Coordinated and Multiple Views in Exploratory Visualization-CMV 2003-*, pp. 26–31. IEEE.
- Brown, A., Kang, J., & Gjestland, T. (2011). Towards standardization in soundscape preference assessment. *Applied Acoustics*, 72(6), 387–392.
- Cai, D., He, X., Li, Z., Ma, W.-Y., & Wen, J.-R. (2004). Hierarchical clustering of www image search results using visual, textual and link information. In *Proceedings of the 12th annual ACM international conference on Multimedia*, pp. 952–959. ACM.
- Caliński, T. & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1), 1–27.
- Cano, P., Kaltenbrunner, M., Gouyon, F., & Batlle, E. (2002). On the use of fastmap for audio retrieval and browsing. In *ISMIR*.

- Carpineto, C., Osiński, S., Romano, G., & Weiss, D. (2009). A survey of web clustering engines. *ACM Computing Surveys (CSUR)*, 41(3), 17.
- Carpineto, C. & Romano, G. (2012). A survey of automatic query expansion in information retrieval. *ACM Computing Surveys (CSUR)*, 44(1), 1.
- Cartwright, M., Seals, A., Salamon, J. et al. (2017). Seeing sound: Investigating the effects of visualizations and complexity on crowd-sourced audio annotations. *Proceedings of the ACM on Human-Computer Interaction*, 1(1).
- Cayton, L. (2008). Fast nearest neighbor retrieval for bregman divergences. In *Proceedings of the 25th international conference on Machine learning*, pp. 112–119. ACM.
- Celma, O., Herrera, P., & Serra, X. (2006). Bridging the music semantic gap. In *Workshop on Mastering the Gap: From Information Extraction to Semantic Representation*, vol. 187. CEUR.
- Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*.
- Choi, J., Lee, J., Park, J., & Nam, J. (2019). Zero-shot learning for audio-based music classification and tagging. *arXiv preprint arXiv:1907.02670*.
- Choi, K., Fazekas, G., Sandler, M., & Cho, K. (2017). Transfer learning for music classification and regression tasks. *arXiv preprint arXiv:1703.09179*.
- Cramer, J., Wu, H.-H., Salamon, J., & Bello, J. P. (2019). Look, listen, and learn more: Design choices for deep audio embeddings. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3852–3856. IEEE.

- Crocco, M., Cristani, M., Trucco, A., & Murino, V. (2016). Audio surveillance: A systematic review. *ACM Computing Surveys (CSUR)*, 48(4), 1–46.
- Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6), 391–407.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 248–255. Ieee.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Do, H. M., Pham, M., Sheng, W., Yang, D., & Liu, M. (2018). Rish: A robot-integrated smart home for elderly care. *Robotics and Autonomous Systems*, 101, 74–92.
- Dong, W., Moses, C., & Li, K. (2011). Efficient k-nearest neighbor graph construction for generic similarity measures. In *Proceedings of the 20th international conference on World wide web*, pp. 577–586. ACM.
- Drossos, K., Lipping, S., & Virtanen, T. (2020). Clotho: an audio captioning dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 736–740. IEEE.
- Drossos, K., Mimitakis, S. I., Serdyuk, D., Schuller, G., Virtanen, T., & Bengio, Y. (2018). Mad twinnet: Masker-denoiser architecture with twin networks for monaural sound source separation. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE.

- Dumoulin, V. & Visin, F. (2016). A guide to convolution arithmetic for deep learning.
- Eades, P. (1984). A heuristic for graph drawing. *Congressus numerantium*, 42, 149–160.
- Eades, P. & Klein, K. (2018). Graph visualization. In *Graph Data Management*, pp. 33–70. Springer.
- Engel, J., Resnick, C., Roberts, A., Dieleman, S., Norouzi, M., Eck, D., & Simonyan, K. (2017). Neural audio synthesis of musical notes with wavenet autoencoders. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1068–1077. JMLR. org.
- Eronen, A. & Klapuri, A. (2000). Musical instrument recognition using cepstral coefficients and temporal features. In *Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on*, vol. 2, pp. II753–II756. IEEE.
- et al., A. V. (2017). Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008.
- et al., D. J. (2019). Improving transformer-based speech recognition using unsupervised pre-training. *arXiv preprint arXiv:1910.09932*.
- Fagan, J. C. (2010). Usability studies of faceted browsing: A literature review. *Information Technology and Libraries*, 29(2), 58.
- Fahad, A., Alshatri, N., Tari, Z., Alamri, A., Khalil, I., Zomaya, A. Y., Foufou, S., & Bouras, A. (2014). A survey of clustering algorithms for big data: Taxonomy and empirical analysis. *IEEE transactions on emerging topics in computing*, 2(3), 267–279.
- Favory, X., Drossos, K., Virtanen, T., & Serra, X. (2020a). COALA: Co-aligned autoencoders for learning semantically enriched audio representations. In *workshop on Self-supervision in Audio and Speech*

at the 37th International Conference on Machine Learning, Vienna, Austria.

- Favory, X., Drossos, K., Virtanen, T., & Serra, X. (2020b). Learning contextual tag embeddings for cross-modal alignment of audio and tags. *arXiv preprint arXiv:2010.14171*.
- Favory, X., Fonseca, E., Font, F., & Serra, X. (2018). Facilitating the manual annotation of sounds when using large taxonomies. In *Proceedings of the 23rd Conference of Open Innovations Association FRUCT*, p. 60.
- Favory, X., Font, F., & Serra, X. (2020c). Search result clustering in collaborative sound collections. In *Proceedings of the 2020 International Conference on Multimedia Retrieval*, pp. 207–214.
- Favory, X. & Serra, X. (2018). Multi web audio sequencer: collaborative music making. In *4th Web Audio Conference*.
- Fonseca, E., Favory, X., Pons, J., Font, F., & Serra, X. (2020a). FSD50K: an open dataset of human-labeled sound events. *preprint arXiv:2010.00475*.
- Fonseca, E., Ortego, D., McGuinness, K., O'Connor, N. E., & Serra, X. (2020b). Unsupervised contrastive learning of sound event representations. *arXiv preprint arXiv:2011.07616*.
- Fonseca, E., Plakal, M., Ellis, D. P., Font, F., Favory, X., & Serra, X. (2019a). Learning sound event classifiers from web audio with noisy labels. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 21–25. IEEE.
- Fonseca, E., Plakal, M., Font, F., Ellis, D. P., Favory, X., Pons, J., & Serra, X. (2018). General-purpose tagging of freesound audio with audioset labels: Task description, dataset, and baseline. *arXiv preprint arXiv:1807.09902*.

- Fonseca, E., Plakal, M., Font, F., Ellis, D. P., & Serra, X. (2019b). Audio tagging with noisy labels and minimal supervision. *arXiv preprint arXiv:1906.02975*.
- Fonseca, E., Pons, J., Favory, X., Font, F., Bogdanov, D., Ferraro, A., Oramas, S., Porter, A., & Serra, X. (2017a). Freesound datasets: A platform for the creation of open audio datasets. In *Proceedings of the International Society for Music Information Retrieval Conference*.
- Fonseca, E., Pons Puig, J., Favory, X., Font Corbera, F., Bogdanov, D., Ferraro, A., Oramas, S., Porter, A., & Serra, X. (2017b). Freesound datasets: a platform for the creation of open audio datasets. In *Hu X, Cunningham SJ, Turnbull D, Duan Z, editors. Proceedings of the 18th ISMIR Conference; 2017 oct 23-27; Suzhou, China.[Canada]: International Society for Music Information Retrieval; 2017. p. 486-93*. International Society for Music Information Retrieval (ISMIR).
- Font, F. & Bandiera, G. (2017). Freesound explorer: Make music while discovering freesound!
- Font, F., Roma, G., & Serra, X. (2013a). Freesound technical demo. In *Proceedings of the 21st ACM international conference on Multimedia*, pp. 411–412.
- Font, F., Roma, G., & Serra, X. (2018). Sound sharing and retrieval. In *Computational Analysis of Sound Scenes and Events*, pp. 279–301. Springer.
- Font, F., Serra, J., & Serra, X. (2013b). Folksonomy-based tag recommendation for collaborative tagging systems. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 9(2), 1–30.
- Fortunato, S. (2010). Community detection in graphs. *Physics reports*, 486(3-5), 75–174.

- Fu, Z., Lu, G., Ting, K. M., & Zhang, D. (2010). A survey of audio-based music classification and annotation. *IEEE transactions on multimedia*, 13(2), 303–319.
- Furnas, G. W., Landauer, T. K., Gomez, L. M., & Dumais, S. T. (1987). The vocabulary problem in human-system communication. *Communications of the ACM*, 30(11), 964–971.
- Galanopoulos, D. & Mezaris, V. (2020). Attention mechanisms, signal encodings and fusion strategies for improved ad-hoc video search with dual encoding networks. In *Proceedings of the 2020 International Conference on Multimedia Retrieval*, pp. 336–340.
- Gaver, W. W. (1993). What in the world do we hear?: An ecological approach to auditory event perception. *Ecological psychology*, 5(1), 1–29.
- Gemmeke, J. F., Ellis, D. P., Freedman, D., Jansen, A., Lawrence, W., Moore, R. C., Plakal, M., & 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)*, pp. 776–780. IEEE.
- Ghias, A., Logan, J., Chamberlin, D., & Smith, B. C. (1995). Query by humming: musical information retrieval in an audio database. In *Proceedings of the third ACM international conference on Multimedia*, pp. 231–236.
- Halkidi, M. & Vazirgiannis, M. (2001). Clustering validity assessment: Finding the optimal partitioning of a data set. In *Proceedings 2001 IEEE International Conference on Data Mining*, pp. 187–194. IEEE.
- Hardoon, D. R., Szedmak, S., & Shawe-Taylor, J. (2004). Canonical correlation analysis: An overview with application to learning methods. *Neural computation*, 16(12), 2639–2664.

- Hassan-Montero, Y. & Herrero-Solana, V. (2006). Improving tag-clouds as visual information retrieval interfaces. In *International conference on multidisciplinary information sciences and technologies*, pp. 25–28.
- Hearst, M. A., Karger, D. R., & Pedersen, J. O. (1995). Scatter/gather as a tool for the navigation of retrieval results. In *Working Notes of the AAAI Fall Symposium on AI Applications in Knowledge Navigation and Retrieval*, Cambridge, MA.
- Helén, M. & Virtanen, T. (2009). Audio query by example using similarity measures between probability density functions of features. *EURASIP Journal on Audio, Speech, and Music Processing*, 2010, 1–12.
- Herrera-Boyer, P., Peeters, G., & Dubnov, S. (2003). Automatic classification of musical instrument sounds. *Journal of New Music Research*, 32(1), 3–21.
- Hershey, S., Chaudhuri, S., Ellis, D. P., Gemmeke, J. F., Jansen, A., Moore, R. C., Plakal, M., Platt, D., Saurous, R. A., Seybold, B. et al. (2017). Cnn architectures for large-scale audio classification. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 131–135. IEEE.
- Hubert, L. & Arabie, P. (1985). Comparing partitions. *Journal of classification*, 2(1), 193–218.
- Hutchinson, H., Mackay, W., Westerlund, B., Bederson, B. B. et al. (2003). Technology probes: inspiring design for and with families. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 17–24. ACM.
- Hüwel, A., Adiloğlu, K., & Bach, J.-H. (2020). Hearing aid research data set for acoustic environment recognition. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 706–710. IEEE.

- Ipeirotis, P. G., Provost, F., & Wang, J. (2010). Quality management on amazon mechanical turk. In *Proceedings of the ACM SIGKDD workshop on human computation*, pp. 64–67.
- Jacobs, P. S. (2014). *Text-based intelligent systems: Current research and practice in information extraction and retrieval*. Psychology Press.
- Jain, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8), 651–666.
- Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3), 264–323.
- Jansen, A., Gemmeke, J. F., Ellis, D. P., Liu, X., Lawrence, W., & Freedman, D. (2017). Large-scale audio event discovery in one million youtube videos. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 786–790. IEEE.
- Janvier, M., Alameda-Pineda, X., Girinz, L., & Horaud, R. (2012). Sound-event recognition with a companion humanoid. In *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, pp. 104–111. IEEE.
- Jensen, J. H., Ellis, D. P., Christensen, M. G., & Jensen, S. H. (2007). Evaluation distance measures between gaussian mixture models of mfccs.
- Jing, F., Wang, C., Yao, Y., Deng, K., Zhang, L., & Ma, W.-Y. (2006). Igroup: web image search results clustering. In *Proceedings of the 14th ACM international conference on Multimedia*, pp. 377–384.
- John, G. H., Kohavi, R., & Pflieger, K. (1994). Irrelevant features and the subset selection problem. In *Machine Learning Proceedings 1994*, pp. 121–129. Elsevier.
- Kereliuk, C., Sturm, B. L., & Larsen, J. (2015). Deep learning and music adversaries. *IEEE Transactions on Multimedia*, 17(11), 2059–2071.

- Kholghi, M., Phillips, Y., Towsey, M., Sitbon, L., & Roe, P. (2018). Active learning for classifying long-duration audio recordings of the environment. *Methods in Ecology and Evolution*, 9(9), 1948–1958.
- Kim, Y. E., Schmidt, E. M., Migneco, R., Morton, B. G., Richardson, P., Scott, J., Speck, J. A., & Turnbull, D. (2010). Music emotion recognition: A state of the art review. In *Proc. ISMIR*, vol. 86, pp. 937–952. Citeseer.
- Law, E., West, K., Mandel, M. I., Bay, M., & Downie, J. S. (2009). Evaluation of algorithms using games: The case of music tagging. In *ISMIR*, pp. 387–392.
- Lawrence, L. (2008). Remix: Making art and commerce thrive in the hybrid economy. *New York*.
- Le-Khac, P. H., Healy, G., & Smeaton, A. F. (2020). Contrastive representation learning: A framework and review. *arXiv preprint arXiv:2010.05113*.
- Lee, J., Park, J., Kim, K. L., & Nam, J. (2018). Samplecnn: End-to-end deep convolutional neural networks using very small filters for music classification. *Applied Sciences*, 8(1), 150.
- Lew, M. S., Sebe, N., Djeraba, C., & Jain, R. (2006). Content-based multimedia information retrieval: State of the art and challenges. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 2(1), 1–19.
- Li, T., Ogihara, M., & Li, Q. (2003). A comparative study on content-based music genre classification. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pp. 282–289. ACM.
- Liu, T., Rosenberg, C., & Rowley, H. A. (2007). Clustering billions of images with large scale nearest neighbor search. In *2007 IEEE*

- workshop on applications of computer vision (WACV'07)*, pp. 28–28. IEEE.
- Liu, Y., Li, Z., Xiong, H., Gao, X., & Wu, J. (2010). Understanding of internal clustering validation measures. In *2010 IEEE International Conference on Data Mining*, pp. 911–916. IEEE.
- Manning, C., Raghavan, P., & Schütze, H. (2010). Introduction to information retrieval. *Natural Language Engineering*, *16*(1), 100–103.
- Manning, C. D., Schütze, H., & Raghavan, P. (2008). *Introduction to information retrieval*. Cambridge university press.
- Marimont, R. & Shapiro, M. (1979). Nearest neighbour searches and the curse of dimensionality. *IMA Journal of Applied Mathematics*, *24*(1), 59–70.
- Marlow, C., Naaman, M., Boyd, D., & Davis, M. (2006). Ht06, tagging paper, taxonomy, flickr, academic article, to read. In *Proceedings of the seventeenth conference on Hypertext and hypermedia*, pp. 31–40. ACM.
- Martinez, E., Celma, O., Sordo, M., De Jong, B., & Serra, X. (2009). Extending the folksonomies of freesound.org using content-based audio analysis. In *Sound and Music Computing Conference, Porto, Portugal*.
- Martins, L. G., Burred, J. J., Tzanetakis, G., & Lagrange, M. (2007). Polyphonic instrument recognition using spectral clustering. In *ISMIR*, pp. 213–218.
- McFee, B., Humphrey, E. J., & Urbano, J. (2016). A plan for sustainable mir evaluation. In *17th International Society for Music Information Retrieval Conference (ISMIR 2016); 2016 Aug 7-11; New York City, USA.[Place unknown]: International Society for Music Information Retrieval; 2016. p. 285-291*. International Society for Music Information Retrieval (ISMIR).

- McFee, B., Raffel, C., Liang, D., Ellis, D. P., McVicar, M., Battenberg, E., & Nieto, O. (2015). librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, vol. 8.
- Mecca, G., Raunich, S., & Pappalardo, A. (2007). A new algorithm for clustering search results. *Data & Knowledge Engineering*, 62(3), 504–522.
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Miller, G. A. (1998). *WordNet: An electronic lexical database*. MIT press.
- Mimilakis, S. I., Drossos, K., Santos, J. F., Schuller, G., Virtanen, T., & Bengio, Y. (2018). Monaural singing voice separation with skip-filtering connections and recurrent inference of time-frequency mask. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 721–725.
- Mitchell, T., Cohen, W., Hruschka, E., Talukdar, P., Yang, B., Betteridge, J., Carlson, A., Dalvi, B., Gardner, M., Kisiel, B. et al. (2018). Never-ending learning. *Communications of the ACM*, 61(5), 103–115.
- Morschheuser, B., Hamari, J., & Koivisto, J. (2016). Gamification in crowdsourcing: a review. In *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pp. 4375–4384. IEEE.
- Muller, M. & Ewert, S. (2010). Towards timbre-invariant audio features for harmony-based music. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(3), 649–662.
- Navigli, R. (2009). Word sense disambiguation: A survey. *ACM computing surveys (CSUR)*, 41(2), 1–69.

- Neumayer, R., Dittenbach, M., & Rauber, A. (2005a). Playsom and pock-etsomplayer, alternative interfaces to large music collections. In *IS-MIR*, pp. 618–623. Citeseer.
- Neumayer, R., Lidy, T., & Rauber, A. (2005b). *Content-based organiza-tion of digital audio collections*. na.
- Newman, M. E. & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical review E*, 69(2), 026113.
- Nielsen, J. (2000). Why You Only Need to Test with 5 Users. *Jakob Nielsen's Alertbox*, 19(September 23), 1–4.
- Niessen, M. E., Van Kasteren, T. L., & Merentitis, A. (2013). Hierarchi-cal modeling using automated sub-clustering for sound event recog-nition. In *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 1–4. IEEE.
- Oord, A. v. d., Li, Y., & Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.
- Osiński, S. (2003). An algorithm for clustering of web search results. *Master, Poznań University of Technology, Poland*.
- Pachet, F. & Cazaly, D. (2000). A taxonomy of musical genres. In *Content-Based Multimedia Information Access-Volume 2*, pp. 1238–1245. Centre de Hautes Etudes Internationale D'Informatique Doc-umentaire.
- Parisi, G. I., Kemker, R., Part, J. L., Kanan, C., & Wermter, S. (2019). Continual lifelong learning with neural networks: A review. *Neural Networks*, 113, 54–71.
- Park, C., Min, C., Bhattacharya, S., & Kawsar, F. (2020). Augmenting conversational agents with ambient acoustic contexts. In *22nd Inter-national Conference on Human-Computer Interaction with Mobile Devices and Services*, pp. 1–9.

- Park, J., Lee, J., Park, J., Ha, J.-W., & Nam, J. (2017). Representation learning of music using artist labels. *arXiv preprint arXiv:1710.06648*.
- Peeters, G., Giordano, B. L., Susini, P., Misdariis, N., & McAdams, S. (2011). The timbre toolbox: Extracting audio descriptors from musical signals. *The Journal of the Acoustical Society of America*, 130(5), 2902–2916.
- Petkos, G., Schinas, M., Papadopoulos, S., & Kompatsiaris, Y. (2017). Graph-based multimodal clustering for social multimedia. *Multimedia Tools and Applications*, 76(6), 7897–7919.
- Pons, J., Nieto, O., Prockup, M., Schmidt, E., Ehmann, A., & Serra, X. (2017a). End-to-end learning for music audio tagging at scale. *arXiv preprint arXiv:1711.02520*.
- Pons, J. & Serra, X. (2019a). musicnn: Pre-trained convolutional neural networks for music audio tagging. *arXiv preprint arXiv:1909.06654*.
- Pons, J. & Serra, X. (2019b). Randomly weighted cnns for (music) audio classification. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 336–340. IEEE.
- Pons, J., Slizovskaia, O., Gong, R., Gómez, E., & Serra, X. (2017b). Timbre analysis of music audio signals with convolutional neural networks. In *2017 25th European Signal Processing Conference (EUSIPCO)*, pp. 2744–2748. IEEE.
- Qi, P., Zhu, X., Zhou, G., Zhang, Y., Wang, Z., Ren, L., Fan, Y., & Gai, K. (2020). Search-based user interest modeling with lifelong sequential behavior data for click-through rate prediction. *arXiv preprint arXiv:2006.05639*.

- Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. In *International Conference on Learning Representations (ICLR)*.
- Raghu, M., Gilmer, J., Yosinski, J., & Sohl-Dickstein, J. (2017). Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, pp. 6076–6085.
- Ramires, A., Chandna, P., Favory, X., Gómez, E., & Serra, X. (2020). Neural percussive synthesis parameterised by high-level timbral features. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 786–790. IEEE.
- Ramires, A. & Serra, X. (2019). Data augmentation for instrument classification robust to audio effects. *arXiv preprint arXiv:1907.08520*.
- Roma, G., Xambó, A., Herrera, P., & Laney, R. (2012). Factors in human recognition of timbre lexicons generated by data clustering.
- Roma Trepát, G. et al. (2015). Algorithms and representations for supporting online music creation with large-scale audio databases.
- Romano, S., Vinh, N. X., Bailey, J., & Verspoor, K. (2016). Adjusting for chance clustering comparison measures. *The Journal of Machine Learning Research*, 17(1), 4635–4666.
- Russakovsky, O., Deng, J., Su, H., Krause, J. et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3), 211–252.
- Sabou, M., Bontcheva, K., Derczynski, L., & Scharl, A. (2014). Corpus annotation through crowdsourcing: Towards best practice guidelines. In *LREC*, pp. 859–866.

- Sadaf, K. & Alam, M. (2012). Web search result clustering-a review. *International Journal of Computer Science and Engineering Survey*, 3(4), 85.
- Salamon, J. & Bello, J. P. (2017). Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, 24(3), 279–283.
- Salamon, J., Jacoby, C., & Bello, J. P. (2014). A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 1041–1044.
- Salton, G. (1968). Automatic content analysis in information retrieval. Tech. rep., Cornell University.
- Salton, G. (1989). Automatic text processing: The transformation, analysis, and retrieval of. *Reading: Addison-Wesley*, 169.
- Saxena, A., Prasad, M., Gupta, A., Bharill, N., Patel, O. P., Tiwari, A., Er, M. J., Ding, W., & Lin, C.-T. (2017). A review of clustering techniques and developments. *Neurocomputing*, 267, 664–681.
- Schaeffer, S. E. (2007). Graph clustering. *Computer science review*, 1(1), 27–64.
- Schafer, R. M. (1993). *The soundscape: Our sonic environment and the tuning of the world*. Simon and Schuster.
- Schedl, M. (2017). Intelligent user interfaces for social music discovery and exploration of large-scale music repositories. In *Proceedings of the 2017 ACM Workshop on Theory-Informed User Modeling for Tailoring and Personalizing Interfaces*, pp. 7–11.
- Schonfeld, E., Ebrahimi, S., Sinha, S., Darrell, T., & Akata, Z. (2019). Generalized zero-and few-shot learning via aligned variational autoencoders. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8247–8255.

- Shuyang, Z., Heittola, T., & Virtanen, T. (2017). Active learning for sound event classification by clustering unlabeled data. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 751–755. IEEE.
- Silberer, C. & Lapata, M. (2014). Learning grounded meaning representations with autoencoders. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 721–732.
- Simonyan, K. & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Sinclair, J. & Cardew-Hall, M. (2008). The folksonomy tag cloud: when is it useful? *Journal of Information Science*, 34(1), 15–29.
- Stefanowski, J. & Weiss, D. (2003). Carrot 2 and language properties in web search results clustering. In *International Atlantic Web Intelligence Conference*, pp. 240–249. Springer.
- Stodden, V., McNutt, M., Bailey, D. H., Deelman, E., Gil, Y., Hanson, B., Heroux, M. A., Ioannidis, J. P., & Tauber, M. (2016). Enhancing reproducibility for computational methods. *Science*, 354(6317), 1240–1241.
- Sturm, B. L. (2013). The gtzan dataset: Its contents, its faults, their effects on evaluation, and its future use. *arXiv preprint arXiv:1306.1461*.
- Suh, D., Lee, K., Lee, J., Park, J., & Nam, J. (2017). Music galaxy hitchhiker: 3d web music navigation through audio space trained with tag and artist labels. In *The 18th International Society for Musical Information Retrieval Conference (ISMIR)*. ISMIR.
- Sun, C., Myers, A., Vondrick, C., Murphy, K., & Schmid, C. (2019). Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 7464–7473.

- Surís, D., Duarte, A., Salvador, A., Torres, J., & Giró-i Nieto, X. (2018). Cross-modal embeddings for video and audio retrieval. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 0–0.
- Tunkelang, D. (2009). Faceted search. *Synthesis lectures on information concepts, retrieval, and services*, 1(1), 1–80.
- Turnbull, D., Barrington, L., Torres, D., & Lanckriet, G. (2008). Semantic annotation and retrieval of music and sound effects. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2), 467–476.
- Turpault, N., Serizel, R., Salamon, J., & Shah, A. P. (2019a). Sound event detection in domestic environments with weakly labeled data and soundscape synthesis.
- Turpault, N., Serizel, R., & Vincent, E. (2019b). Semi-supervised triplet loss based learning of ambient audio embeddings. In *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.
- Tzanetakis, G. & Cook, P. (2001). Marsyas3d: a prototype audio browser-editor using a large scale immersive visual and audio display. Georgia Institute of Technology.
- Tzanetakis, G. & Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on speech and audio processing*, 10(5), 293–302.
- Urbain, G., Frisson, C., Moinet, A., & Dutoit, T. (2016). A semantic and content-based search user interface for browsing large collections of foley sounds. In *Proceedings of the Audio Mostly 2016*, pp. 272–277.
- Van Den Oord, A., Dieleman, S., & Schrauwen, B. (2014). Transfer learning by supervised pre-training for audio-based music classification. In *Conference of the International Society for Music Information Retrieval (ISMIR 2014)*.

- Van Den Oord, A., Vinyals, O. et al. (2017). Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, pp. 6306–6315.
- Vinh, N. X., Epps, J., & Bailey, J. (2010). Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research*, 11(Oct), 2837–2854.
- Weinberger, K. Q. & Saul, L. K. (2009). Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10(2).
- Wisdom, S., Erdogan, H., Ellis, D., Serizel, R., Turpault, N., Fonseca, E., Salamon, J., Seetharaman, P., & Hershey, J. (2020). What’s all the fuss about free universal sound separation data? *arXiv preprint arXiv:2011.00803*.
- Won, M., Oramas, S., Nieto, O., Gouyon, F., & Serra, X. (2020). Multi-modal metric learning for tag-based music retrieval. *arXiv preprint arXiv:2010.16030*.
- Wongsuphasawat, K., Moritz, D., Anand, A., Mackinlay, J., Howe, B., & Heer, J. (2015). Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE transactions on visualization and computer graphics*, 22(1), 649–658.
- Wu, Y., Wang, S., Song, G., & Huang, Q. (2019). Learning fragment self-attention embeddings for image-text matching. In *Proceedings of the 27th ACM International Conference on Multimedia*.
- Xu, D. & Tian, Y. (2015). A comprehensive survey of clustering algorithms. *Annals of Data Science*, 2(2), 165–193.
- Xu, K., Cai, H., Liu, X., Gao, Z., & Zhang, B. (2017). North atlantic right whale call detection with very deep convolutional neural networks. *The Journal of the Acoustical Society of America*, 141(5), 3944–3945.

- Xu, R. & Wunsch, D. C. (2005). Survey of clustering algorithms.
- Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pp. 3320–3328.
- Zamir, O. & Etzioni, O. (1999). Grouper: a dynamic clustering interface to web search results. *Computer Networks*, 31(11), 1361–1374.

Publications by the author

Under review

Favory, X., Drossos, K., Virtanen, T., & Serra, X. (2020b). Learning contextual tag embeddings for cross-modal alignment of audio and tags. *arXiv preprint arXiv:2010.14171*

Fonseca, E., Favory, X., Pons, J., Font, F., & Serra, X. (2020a). FSD50K: an open dataset of human-labeled sound events. *preprint arXiv:2010.00475*

Conference papers

Favory, X., Drossos, K., Virtanen, T., & Serra, X. (2020a). COALA: Co-aligned autoencoders for learning semantically enriched audio representations. In *workshop on Self-supervision in Audio and Speech at the 37th International Conference on Machine Learning, Vienna, Austria*.

Favory, X., Font, F., & Serra, X. (2020c). Search result clustering in collaborative sound collections. In *Proceedings of the 2020 International Conference on Multimedia Retrieval*, pp. 207–214

Ramires, A., Chandna, P., Favory, X., Gómez, E., & Serra, X. (2020). Neural percussive synthesis parameterised by high-level timbral features. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 786–790. IEEE

Fonseca, E., Plakal, M., Ellis, D. P., Font, F., Favory, X., & Serra, X. (2019a). Learning sound event classifiers from web audio with noisy labels. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 21–25. IEEE

Favory, X., Fonseca, E., Font, F., & Serra, X. (2018). Facilitating the manual annotation of sounds when using large taxonomies. In *Proceedings of the 23rd Conference of Open Innovations Association FRUCT*, p. 60

Favory, X. & Serra, X. (2018). Multi web audio sequencer: collaborative music making. In *4th Web Audio Conference*

Fonseca, E., Pons, J., Favory, X., Font, F., Bogdanov, D., Ferraro, A., Oramas, S., Porter, A., & Serra, X. (2017a). Freesound datasets: A platform for the creation of open audio datasets. In *Proceedings of the International Society for Music Information Retrieval Conference*

