FSD50K: An Open Dataset of Human-Labeled Sound Events

Eduardo Fonseca, *Student Member, IEEE,* Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra

Abstract—Most existing datasets for sound event recognition (SER) are relatively small and/or domain-specific, with the exception of AudioSet, based on over 2M tracks from YouTube videos and encompassing over 500 sound classes. However, AudioSet is not an open dataset as its official release consists of pre-computed audio features. Downloading the original audio tracks can be problematic due to YouTube videos gradually disappearing and usage rights issues. To provide an alternative benchmark dataset and thus foster SER research, we introduce *FSD50K*, an open dataset containing over 51k audio clips totalling over 100h of audio manually labeled using 200 classes drawn from the AudioSet Ontology. The audio clips are licensed under Creative Commons licenses, making the dataset freely distributable (including waveforms). We provide a detailed description of the FSD50K creation process, tailored to the particularities of Freesound data, including challenges encountered and solutions adopted. We include a comprehensive dataset characterization along with discussion of limitations and key factors to allow its audio-informed usage. Finally, we conduct sound event classification experiments to provide baseline systems as well as insight on the main factors to consider when splitting Freesound audio data for SER. Our goal is to develop a dataset to be widely adopted by the community as a new open benchmark for SER research.

Index Terms—audio dataset, sound event, recognition, classification, tagging, data collection, environmental sound.

I. INTRODUCTION

SOUND event recognition (SER) is the task of automatically identifying the sounds occurring in our daily lives, assigning a label within a target set of sound classes.[1] SER has gained increasing attention in the past few years, becoming a key component in applications related to healthcare [1]–[3], urban sound planning [4], bioacoustics monitoring [5]–[7], multimedia event detection [8], large-scale event discovery [9], surveillance [10], [11], or noise monitoring for industrial applications [12]. The SER research community has grown substantially over the last decade, as evidenced by the increasing traction of the *Detection and Classification of Acoustic Scenes and Events* (DCASE) Challenge and Workshop [13], which promote research and evaluation on common publicly available datasets. Early stage works in SER relied on feature engineering approaches using standard machine learning classifiers such as support vector machines [14], Gaussian mixture models [15] or matrix factorization techniques [16]. This initial trend was followed by the rapid adoption of deep learning approaches using fully connected neural networks [17], convolutional neural networks (CNN) [18], recurrent neural networks (RNN) [19], or combinations thereof [20]. To allow successful exploitation of the data hungry deep learning approaches, it became evident the need for new, larger, and more comprehensive data resources for development and evaluation of SER models. In contrast, previous SER datasets were of a more limited size and coverage (e.g. [21]–[23]). In the current paradigm, datasets in SER are crucial, similarly as in computer vision [24], as exemplified by the significant breakthroughs that ImageNet has allowed for image recognition [25].

To address the lack of large datasets in SER, AudioSet was released in 2017. AudioSet consists of $\approx$2.1M audio clips manually labeled using 527 classes [26]. Its unprecedented size, coverage and diversity represented a milestone that has transformed SER research. However, in our view, AudioSet has the major shortcoming of not being an open dataset, as we explain next. Specifically, AudioSet is composed of audio tracks taken from YouTube videos, which are not freely distributable due to YouTube Terms of Service. This is the reason why AudioSet is released as a dataset of audio features (instead of audio waveforms),[2] which are extracted at a time resolution of 960ms using a pre-trained model. This limits the adoption and flexibility of a number of SER methods. For this reason, some researchers opt to download and use the audio tracks from the original YouTube videos, despite the intrinsic issues entailed in this process. These issues include the burden of downloading a massive amount of data from a non-official release, and the fact that the constituent videos are gradually disappearing. More specifically, videos can become unavailable due to a variety of reasons such as deletions of videos or user accounts, privacy issues, copyright claims, or country-dependant availability. In an attempt to download the AudioSet audio tracks, we could download 18,205 from 20,371 evaluation segments, and 19,862 from 22,160 balanced train segments—a loss of 10.6% and 10.4% respectively.[3] The fact that the amount of evaluation and train clips available decreases over time with non-negligible differences limits AudioSet suitability for systems' benchmarking.

After AudioSet, several efforts in dataset creation for SER have been made (e.g. [27]–[34]). Nonetheless, these recent datasets are task/domain-specific, or of a much more limited coverage (e.g., usually featuring few tens of classes), and some of them are composed of synthetic audio material. This contrasts with the computer vision field, where major efforts have been made to collect large and general-purpose

The authors are with the Music Technology Group at Universitat Pompeu Fabra, Barcelona, Spain. e-mail for all authors: (name.surname@upf.edu).

[1]We shall use the expression sound event *recognition* broadly to encompass both sound event *classification* or *tagging* (SET)—a task requiring to identify what sound event classes are present in an audio clip, regardless of start time and end time—as well as sound event *detection* (SED)—a task requiring to localize and identify sound events in an audio clip with start and end times.

[2]https://research.google.com/audioset/download.html

[3]Data from May 11th, 2020.

datasets as alternatives to ImageNet (e.g. [35]–[37]), allowing benchmarking on complementary recognition problems. Thus, the SER field lags far behind in terms of dataset availability, and we believe that open and sustainable dataset creation initiatives are needed to foster SER research and, more generally, machine listening research. In addition, we think it is important to document at length the main aspects of data collection and curation when releasing a dataset—a common practice in computer vision [25], [37] that has also recently been proposed in audio research [38]. Making this information available allows researchers to incorporate data-informed decisions in the design of learning pipelines and in the analysis of results, and can also serve as inspiration for potential dataset creators.

To address these issues and foster SER resesarch, in this paper we introduce *FSD50K* (**Free**sound **D**ataset **50k**): a dataset containing 51,197 audio clips totalling over 100h of audio manually labeled using 200 classes drawn from the AudioSet Ontology. The audio clips are gathered from Freesound and are licensed under Creative Commons (CC) licenses, which allow easy sharing and reuse, thereby making the dataset freely distributable (including audio waveforms). To our knowledge, this is the largest fully-open dataset of human-labeled sound events, and the second largest after AudioSet.

A. Contributions

Our contributions are as follows:
1) a human-labeled open dataset primarily designed for the development and evaluation of multi-label sound event classification systems, but that also allows a variety of sound event research tasks,
2) a detailed description of the FSD50K creation process tailored to the particularities of Freesound data, including challenges encountered and solutions adopted (Sec. III),
3) a comprehensive characterization of the dataset along with discussion of limitations and key factors to allow its audio-informed usage (Sec. IV), and
4) a set of sound event classification experiments to provide baseline systems as well as insight on the main factors to consider when splitting Freesound audio data for SER tasks (Sec. V).

The information here presented is useful to researchers using FSD50K (and in general using Freesound data for machine learning) as it allows making data-informed decisions for design choices of machine listening systems. It may also be useful for researchers working on the creation of large-vocabulary datasets. In addition to the audio waveforms and ground truth, FSD50K includes metadata used during the creation process as well as Freesound metadata for the clips forming the dataset (Sec. IV-A). All these resources can be downloaded from Zenodo.[4] Likewise, code[5] for baseline experiments and a companion site[6] for FSD50K is also available. The companion site allows exploring the audio content of FSD50K as well as reporting labelling errors.

---

[4]https://doi.org/10.5281/zenodo.4060432
[5]https://github.com/edufonseca/FSD50K_baseline
[6]https://annotator.freesound.org/fsd/release/FSD50K/

II. RELATED WORK

This Section discusses the most important datasets for sound event tagging (SET),[1] listed in Table I. The datasets are selected based on number of Google Scholar citations, as well as popularity and/or size for the most recent ones. For comparison, the proposed FSD50K is listed at the bottom. The basic common aspect in SET datasets is that labels are provided at the clip-level (without timestamps), usually regarded as *weak* labels. This contrasts with sound event detection (SED) datasets, where sound events are labeled using also start and end times (usually regarded as *strong* labels). We denote *multi-class* datasets (*m-c*) as datasets where each example is labeled with only one class label, which are sometimes denoted as *single-label datasets*. In a *multi-label* dataset (*m-l*), in contrast, each example can be labeled with *one or more* class labels.

A. Datasets Released Before AudioSet

Before the release of AudioSet, the most widely used datasets for SET have been UrbanSound8K [39], ESC-50 [22], and to a lesser extent CHiME-home [21]. All of them feature short audio chunks and a total duration of less than 10h. Curiously, the two former are one of the few multi-class balanced datasets in SET—most datasets are unbalanced and/or multi-label—and also the most widely used (besides AudioSet). UrbanSound8K and CHiME-home count with a significant amount of clips per class; nonetheless, part of this abundance comes from the fact that many clips are actually time slices coming from the same original recording. For example, UrbanSound8K is sourced from 1302 Freesound clips. ESC-50 features a large vocabulary (50 classes) when compared to other datasets from 2014/2015, but it suffers from data scarcity (only 40 clips/class). Common to all mentioned datasets is that they provide a k-fold cross validation setup—a practice that tended to disappear after the AudioSet release.

B. AudioSet

Google's AudioSet is the largest dataset of sound events released to date, consisting of ≈2.1M audio clips manually labeled using 527 classes of the AudioSet Ontology [26]. AudioSet is the first dataset to put emphasis on general-purpose SER, enabling sound event recognizers to describe a large variety of sound classes, thus aiming at the transcription of most everyday sounds. AudioSet is split into a train and an evaluation set, and it is highly imbalanced, with some classes being particularly common (e.g. *Music* and *Speech*) while others are much more scarce (e.g. *Toothbrush*). The public release provides a balanced train partition of 22,176 clips in addition to the full unbalanced train set. While the dataset is manually labeled in full, its unprecedented size and coverage comes at the expense of a less precise labeling. The amount of labeling error is estimated at above 50% for ≈18% of the classes.[10] Recently, strong labels for a small portion of AudioSet (≈81k clips) were released [40].

---

[7]http://spandh.dcs.shef.ac.uk//projects/chime/
[8]https://multimediacommons.wordpress.com/yfcc100m-core-dataset/
[9]https://wp.nyu.edu/sonyc/
[10]See https://research.google.com/audioset/dataset/index.html for details on how the quality is estimated, accessed 25th June 2020.

TABLE I
A SELECTION OF MOST RELEVANT DATASETS FOR SET. *m-c* AND *m-l* CORRESPOND TO MULTI-CLASS AND MULTI-LABEL

dataset	clips	clip length	duration	classes	task	source	domain/task
UrbanSound8K [39] (2014)	8732	≤4s	8.8h	10 bal	m-c	Freesound	urban sounds
ESC-50 [22] (2015)	2000	5s	2.8h	50 bal	m-c	Freesound	
CHiME-home [21] (2015)	6138	4s	6.8h	7 unbal	m-l	CHiME[7]	domestic sounds
AudioSet [26] (2017)	≈2.1M	≈10s	≈5731h	527 (un)bal	m-l	YouTube	
FSDnoisy18k [32] (2019)	18,532	0.3-30s	43h	20 unbal	m-c	Freesound	noisy labels
FSDKaggle2019 [33] (2019)	29,266	0.3-30s	103h	80 unbal	m-l	Freesound/YFCC100M[8]	noisy labels & domain mismatch
SONYC-UST-V2 [34] (2020)	18,510	10s	51h	31 unbal	m-l	SONYC[9]	urban sounds
FSD50K (2020)	51,197	0.3-30s	108h	200 unbal	m-l	Freesound	

C. Datasets Released After AudioSet

After AudioSet, some of the released datasets for SET are task-dependent, designed to enable the study of particular problems. Examples include FSDnoisy18k [32] or FSDKaggle2019 [33], focused on learning in conditions of noisy labels and/or acoustic mismatch. Other datasets are domain-specific, with a vocabulary focused on a specific scope, such as SONYC-UST-V2 for urban sounds [34]. Compared to pre-AudioSet datasets, these are slightly larger, especially in terms of duration as they feature longer clips (sometimes of variable length), but also in terms of vocabulary. In addition, they are unbalanced, and the default data split transitioned to a development/evaluation (or train/test) separation. Lastly, a recent large-vocabulary dataset with a substantial amount of data is VGGSound [41], an audio-visual dataset consisting of ≈200k video clips from YouTube encompassing 300 classes. However, VGGSound presents several shortcomings for SET. The focus is put on audio-visual correspondence since the dataset is created mostly through automatic computer vision techniques—hence some classes have a clear visual connotation, e.g., *people eating noodle*. Also, while the dataset is singly-labeled (one machine-generated label per clip), the authors recognize that clips can contain a mixture of sounds. Missing labels are a form of label noise found to impact sound recognizers [42]. While measures can be taken to mitigate their effect on training, in evaluation they can lead to misleading results—an issue that we specifically address in FSD50K (Sec. III-G). In addition, VGGSound suffers from the intrinsic problems of being based on YouTube (Sec. I).

To our knowledge, all datasets in Table I are labelled manually (except a portion of FSDnoisy18k and FSDKaggle2019 purposely included for the study of noisy labels). FSD50K is a superset of the human-labeled portions of FSDnoisy18k and FSDKaggle2019, except for a few audio clips that have been discarded during FSD50K's curation process.

III. DATASET CREATION

A. Design Criteria

As design criteria, we set three basic goals and another three specific goals. The basic goals are: *i)* the dataset must be open and fully distributable, *ii)* it must contain a large vocabulary of everyday sounds, and *iii)* it must be expandable in terms of data and vocabulary. To fulfil these basic goals, we turn to Freesound as a source of data, and to the AudioSet Ontology as a vocabulary to organize the data. Not only do these resources feature a large amount of data and classes, respectively, but Freesound is constantly growing through user uploads, and

the ontology is large and was designed to be expandable. In addition, we set three specific goals related to the labeling of the dataset and to the emphasis put on the evaluation set.

1) Weak Labels: The motivation to label FSD50K with weak labels (i.e., at the clip-level) is that gathering them is simpler, less time consuming and less ambiguous than determining events' onset/offset (i.e., strong labels). Weakly supervised learning has demonstrated effectiveness to learn sound event recognizers, both for classification and detection [43]. Nonetheless, using weak labels imply certain limitations on training and evaluation, which we highlight in Sec. IV-B.

2) Label Quality and Dataset Size: The sound event recognition (SER) field has witnessed a transition away from small and exhaustively labeled datasets (e.g., [21, 22, 39]), in favour of larger datasets that inevitably include less precise labelling, such as AudioSet [26]. This occurs mainly because it is not feasible to exhaustively annotate large amounts of sound event data. In our case, we want to seek a trade-off by prioritizing label quality while ensuring a certain amount of data. Yet, label noise problems also appear in FSD50K, as in any sound event dataset of a certain size (Sec. IV-C).

3) Emphasis on Evaluation Set: This is perhaps the design criteria that mostly determines the creation of FSD50K. An evaluation set defines the target behavior in a recognition task, which makes it possibly the most critical part of a dataset. Consequently, having a comprehensive, diverse, reliably annotated, and real-world representative evaluation set is important for meaningful systems' benchmarking. The importance of reliable evaluation sets is highlighted by recent research in computer vision which focuses on improving the evaluation and/or validation sets of widely-used datasets—[44] for CIFAR-10/-100 [45]; and [46, 47] for ImageNet [48]. In addition, alternative learning paradigms to the traditional supervised learning (using reliably-labelled datasets) start to be promising nowadays. In particular, significant progress is being made in the development of sound event recognizers with noisy supervision [32, 42] or self-supervision [49]–[51]. While these alternatives can minimize the problems of labelling inaccuracies in the development set, or the need for a labeled development set at all, a carefully curated evaluation set is still critical for benchmarking. Relatedly, abundant data resources for training are already available, either from AudioSet, or directly from web audio repositories such as Freesound or Flickr (provided appropriate learning strategies are used). By contrast, to our knowledge, large-vocabulary, carefully-curated evaluation benchmarks are rare—the most prominent being AudioSet's evaluation set, which suffers from issues of label
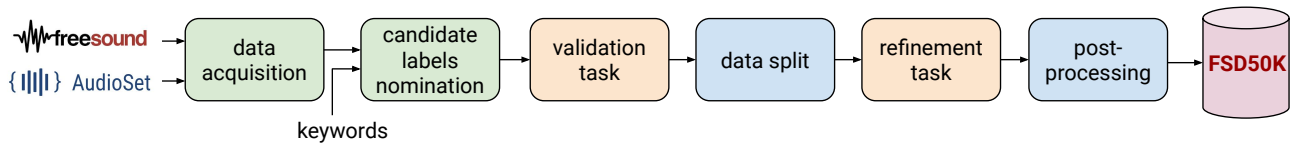
Fig. 1. Overall process of the creation of FSD50K. The process starts from Freesound and the AudioSet Ontology. Stages in green involve automatic data mining, stages in orange correspond to manual annotation tasks, and stages in blue involve data processing to shape the dataset.

noise, stability and/or openness (Secs. I and II). By prioritizing the curation of the evaluation set, we contribute to fill this gap.

To tackle the task of labelling the dataset, two approaches are considered: *i)* manual annotation and *ii)* semi-automatic methods based on Active Learning (AL). Manual annotation is the conventional approach to dataset labelling, as done in AudioSet [26] or ImageNet [48]. While this option is laborious and time consuming, when done properly, we believe it leads to more reliable results than involving automatic methods in the annotation loop. As an alternative, AL aims at maximizing performance with limited labelling budget by selecting the most informative data for the model to learn. Usually, AL is based on an iterative process involving humans in the loop where automatic methods select the samples to annotate. Often, portions of unlabeled data are automatically labelled via propagation of human-provided labels to similar examples, or with semi-supervised learning approaches. Recent works studying AL for SER [52]–[55] report reduced annotation effort with good performance which, in principle, makes AL appealing for dataset creation. However, these works focus on recognition tasks with less than a dozen classes, and most of them deal with single-label classification and use pre-labeled datasets, where the human annotation step is simulated by a simple assignment of the existing ground truth. Extending the methods to a setting like ours is considered out of the scope of this work (albeit an interesting topic for future research).

In order to obtain a high-quality labelling, and being aware of the amount of data to annotate and the budget available, we decide to annotate the dataset manually, similarly as done with AudioSet. While this means a higher human effort, it presents two advantages. First, manually annotating FSD50K gives us a deeper insight into the data that would not have been gained otherwise. Second, it allows us to have a greater control of the labels gathered, as well as to specify not only the labels but also an estimate of sound predominance (Sec. III-E). Obtaining a set of labels as reliable as possible for this first release is a more favorable starting point for potential future expansions, which could rely on (semi-) automatic methods to scale up more efficiently at the expense of label noise.

B. Overall procedure

The overall process of the creation of FSD50K is illustrated in Fig. 1, starting from Freesound and the AudioSet Ontology, and ending with FSD50K. In every intermediate stage, we progressively filter out a quantity of audio clips and classes in the vocabulary. Each stage is described in the next subsections.

C. Data acquisition

The starting point for the creation of FSD50K is an abundant source of audio clips, a vocabulary to annotate them, and

an infrastructure where they can be loaded and annotation tasks can be carried out. These items correspond to Freesound, AudioSet Ontology and Freesound Annotator respectively.

1) Freesound: Freesound[11] is an online collaborative audio clip sharing site [56], with more than 10 million registered users and over 500,000 audio clips.[12] Audio clips shared in Freesound cover a wide variety of audio content, from music samples to environmental sounds, human sounds or audio effects, to name a few. In addition, the users who upload the clips also provide metadata, e.g., a title, several tags (at least three per clip), and textual descriptions. We use the user-provided tags in the creation of FSD50K (Sec. III-D). Since Freesound is collaboratively contributed, it is also very heterogeneous in terms of data origin, recording equipment, and acoustic conditions. All of the content is CC-licensed, which conveniently allows distribution and reuse. Several datasets containing Freesound audio have been widely used by the research community [22, 32, 33, 39, 57, 58], showing its usefulness for dataset creation.

2) AudioSet Ontology: AudioSet Ontology consists of 632 sound event classes arranged in a hierarchy with a maximum depth of 6 levels [26].[13] The set of classes covers a diverse range of everyday sounds, from human and animal sounds, to natural, musical or miscellaneous sounds. Within these main sound families, the content covered includes several facets. The predominant classes correspond to sound events produced by physical sound sources, but there are also some generated by sound production mechanisms (e.g., deformation or impact of materials), and other classes that do not correspond to sound events. Each entry in the ontology includes a textual description among other fields.[14] Note that the AudioSet vocabulary is a subset of 527 classes drawn from the ontology [26]. We use the ontology because it is the most comprehensive vocabulary of everyday sounds available, which makes it convenient to cover Freesound's heterogeneity. Yet, upon careful inspection of the ontology, we realize that improvements could be made in order to make it more suitable for organization of Freesound. However, this task is left out of the scope of this work. For FSD50K, we focus on a subset of the ontology oriented to most common physical sources, and less oriented to ambiguous or less represented classes in common everyday situations. Appendix A clarifies relevant ontology-related nomenclature used in this paper (such as *leaf* or *intermediate* nodes).

---

[11]https://freesound.org/
[12]Data from August 1st, 2021.
[13]It can be explored at https://research.google.com/audioset/ontology/index.html. Sometimes we shall refer to the AudioSet Ontology as *the ontology*.
[14]https://github.com/audioset/ontology

3) Freesound Annotator: Freesound Annotator[15] (FSA) is a website that allows the collaborative creation of open audio datasets based on Freesound content. It serves mainly two goals: the management and exploration of datasets, and the creation and verification of annotations. Originally released on 2017 [59], Freesound Annotator has been the object of continuous development. It started by providing basic prototypes for exploring a taxonomy of audio classes and validating automatically generated annotations. Additional features were incorporated progressively, including annotation tools and quality control mechanisms (see Secs. III-E and III-G). Monitoring tools allow inspection of a dataset progress as well as debugging capabilities. FSA is an open-source project.[16]

D. Candidate Labels Nomination

We started building FSD50K by automatically populating the classes of the ontology with a number of candidate audio clips from Freesound. Candidate clips were selected by matching user-provided tags in Freesound to a set of keywords associated with every class. The goal was to automatically compile a list of candidate labels per clip, indicating potential presence of sound events. The process consisted of two steps.

First, we compiled a list of keywords for almost every class. These are terms related to the class label that are likely to be provided by Freesound users as tags when describing audio clips. Suitable keywords were determined by considering class names and descriptions provided in the ontology, and obtaining the most frequent Freesound tags that co-occur with each target class label. After compiling a first version of the per-class keywords, we manually identified a few classes with very low precision due to pathological inclusion of false positives (e.g., in the *Turkey* class many clips were recordings made in the Eurasian country, instead of containing sounds of the large bird). To minimize this issue, a refinement process was performed by blocking some tags (e.g., "turkish" or "Istanbul" for *Turkey*). As an example, the keywords for the *Meow* class are: "meow", "meowing", "mew", "miaow", and "miaou".

Second, each class was automatically populated with the corresponding Freesound clips. We use the compiled lists of keywords as a mapping between clips in Freesound and class labels in the ontology. Thus, for each clip, all user-provided tags are examined and, when a tag matches a keyword, the clip becomes a candidate clip for the dataset, and the corresponding class label is nominated as a candidate label for the clip. This process was done by using the Freesound API.[17] We employ the Porter Stemming algorithm for term normalisation to make our matching process more robust [60].

In this way we were able to map more than 300,000 Freesound clips to the AudioSet classes. We decided to filter out clips longer than 90s to avoid very large audio clips (this length limit will be further reduced later on, see Sec. III-E). No other filters were applied at this stage. This left us with a total of 268,261 clips with an average of 2.62 candidate labels. This label nomination system induces potential errors as

---

it depends on factors such as class ambiguity and, especially, the choices of Freesound users when providing tags. However, it has the advantage of allowing easy and rapid retrieval for a large variety of classes without training any classifiers.

The outcome of this stage is a list of automatically-generated candidate labels per clip, indicating the potential presence of sound events.

E. Validation Task

The goal of this stage is to manually validate the candidate labels nominated in the previous stage.

1) Initial Prototype of the Annotation Tool: To this end, we designed an annotation tool that was deployed in FSA. Essentially, human raters are presented with a number of audio clips and, for each clip, they must assess the presence of a given sound class. For each class, the annotation process consists of two phases. First, a **training phase** where raters get familiar with the class by looking at its location in the AudioSet Ontology hierarchy, the provided textual description, and representative sound examples. Then, a **validation phase**, in which raters are presented with a series of audio clips from that class (up to 72 clips in 6 *pages* of 12 clips) and prompted the question: *Is <class> present in the following sounds?*. (Fig. 3 shows 3 clips within a page of the final prototype). In this initial prototype, raters must select among "Present", "Not Present", and "Unsure", similarly as done in [26]. Along with an audio player and its waveform, links to each clip's Freesound page were made available, where the original tags and descriptions could be inspected to aid the process.

2) Internal Quality Assessment (IQA): We used the initial prototype to run an Internal Quality Assessment (IQA) with the goal of *i)* assessing the quality of the candidates produced by the nomination system, and *ii)* collecting feedback about the prototype and annotation task for improvements. The IQA consisted of validating 12 candidates (1 page of clips) for every class, covering all classes available. It was carried out by 11 subjects that volunteered to participate, who could leave per-class comments through a text box. The feedback collected in the IQA revealed that the annotation task is complex due to factors such as ambiguity in some class descriptions or the difficulty of annotating sound events with very high inter- and intra-class variation.

3) Final Prototype of the Annotation Tool: Based on the insight from the IQA, we designed the final annotation tool (Figs. 2 and 3) which incorporates the following improvements with respect to the previous version:

- Some AudioSet class descriptions were found to be ambiguous, allowing multiple interpretations and generating doubts as to the class scope. We decided to include a list of **Frequently Asked Questions (FAQs)** in each class description to help homogenize raters' judgment and gather more consistent annotations (see Fig. 2). The full FAQ list is provided with the dataset.
- In some audio clips, several sound events co-existed with different predominance or salience levels, making the "Present" response rather ambiguous for raters. To address this issue, we decided to **split the "Present" response** into "Present and predominant" (PP) and "Present
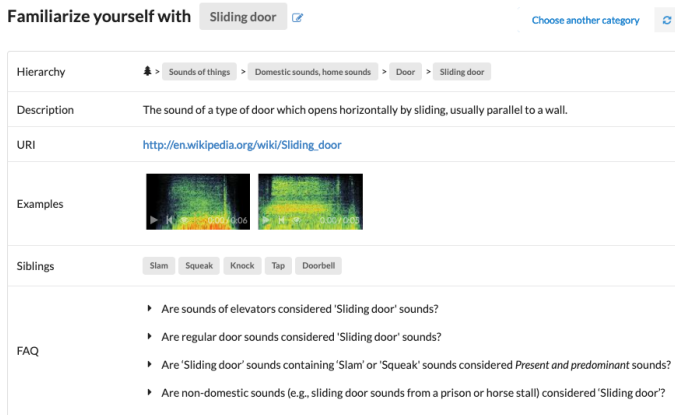
Fig. 2. Screenshot of the "Training phase" page used for the validation task.
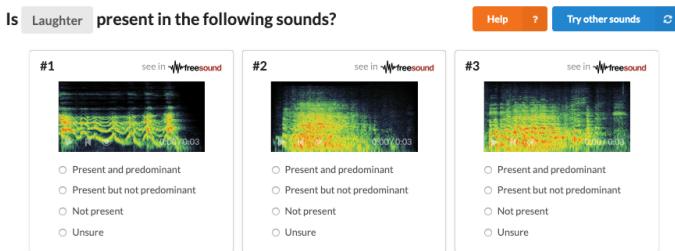

Fig. 3. Screenshot of the "Validation phase" used for the validation task.

but not predominant" (PNP), as specified in Table II.[18] A similar approach was used by Salamon et al. [39]. The main motivation is to ease the annotation task by mitigating a systematic doubt. As an additional benefit, this distinction allows to separate a subset of clips containing mostly isolated and clean sound events (PP ratings) vs. others featuring events from several classes and/or in more adverse acoustic conditions (PNP ratings). This could allow defining robustness tasks such as training or evaluating with a subset of data of more adverse conditions, similarly as done in [61] for sound event detection (SED) or with ImageNet-A for image recognition [62]. Further, the PP/PNP distinction can be useful for source separation studies [63]. We note, however, that this distinction is subjective and these ratings should be used as a rough indication.

- To automatically assess the reliability of the submitted responses, we added **quality control mechanisms** such as the periodic inclusion of verification clips. Whenever the response for one of these clips is wrong, the responses submitted in a given time span are discarded—a common practice in crowdsourcing platforms.
- To further ensure high quality annotations, we decided to require **inter-annotator agreement**. More specifically, each candidate label is presented to several raters until two different raters agree on a response type. Once an inter-annotator agreement is reached, the label is considered as *ground-truth* and it is no longer presented to other raters. A similar practice is done in [21, 27].
- To facilitate the localisation and recognition of sound

events within the audio clips, we added **spectrogram visualizations**, thereby easing the annotation task [64] (the initial prototype featured less-informative waveforms).
- Some audio clips can present highly variable loudness, which can be burdensome for the rater and may affect annotation quality. To mitigate this problem, we **normalize the loudness** of the sound files following the recommendation EBU R-128 [65].
- To select which audio clips to present to each rater, we adopt a **prioritization scheme** that ranks clips according to two criteria: *i)* previously rated label-clip pairs that have not yet reached inter-annotator agreement are prioritized to obtain ground truth labels; *ii)* short clips are prioritized over long ones as shorter clips have a higher label density—considered more informative for learning.

Beyond these improvements, we took two additional measures to improve annotation efficiency. First, the selection of candidates in a number of classes had a very low precision possibly due to sub-optimality of the nomination system. Thus, we decided to discard classes with a rate of "Not Present" responses above 75%, as well as classes with very few candidates and others deemed highly ambiguous for annotation. This left a total of 395 sound classes (a reduction of ≈35%). Second, participants reported that the initial duration limit of 90s was too long for human validation, and a potential cause for fatigue. In addition, the supervision given by weak labels applied to such large lengths is rather vague. Therefore, we decided to discard clips longer than 30s.

4) Annotation Campaign: With the final annotation tool, we launched an annotation campaign to validate the candidate labels at scale. Given that some classes were found to be much more difficult to annotate than others, we decided to gather annotations using both crowdsourcing and hired raters. We divided the classes according to an estimated level of difficulty, based on feedback from the IQA. Table III lists the annotation strategies adopted for each subset of classes. Crowdsourcing consists of gathering validations contributed by any voluntary participant. We made the classes of easy and medium difficulty publicly accessible from FSA,[19] which was promoted in Freesound forums and social media. The most difficult classes, where annotation experience was important to provide reliable responses, were kept private. They were validated by a pool of hired raters who also complemented the crowdsourcing validations in the rest of the classes.

In total, over 350 raters contributed, including voluntary participants, six hired raters, and the first three authors of this paper. The hired raters were subjects with background in audiovisual engineering, including mostly MSc and PhD students from our group, with self-reported healthy hearing. We opted for a small pool of raters in order to have more control over the annotation process and to obtain annotations that are as consistent as possible. We recognize this may induce a certain bias, but we rather have consistent annotations with agreed bias than a certain lack of consistency likely resulting from crowdsourcing annotations for the difficult classes (i.e., label noise). To this end, the hired raters were trained

---

[18]Hereafter, we shall use "Present" to refer to the union of PP and PNP.

[19]https://annotator.freesound.org/fsd/annotate/

TABLE II
RESPONSE TYPES FOR THE VALIDATION TASK

Response type	Meaning
Present and predominant (PP)	The type of sound described is **clearly present** and **predominant**. This means there are no other types of sound, with the exception of low/mild background noise.
Present but not predominant (PNP)	The type of sound described is **present**, but the audio clip also **contains other salient types of sound and/or strong background noise**.
Not Present (NP)	The type of sound described is **not present** in the audio clip.
Unsure (U)	**I am not sure** whether the type of sound described is present or not.

TABLE III
ANNOTATION STRATEGIES IN THE VALIDATION TASK

class difficulty	example	classes	annotation strategy
easy	*Bark*	77	crowdsourcing & hired annotators
medium	*Piano*	100	crowdsourcing & hired annotators
difficult	*Tearing*	218	hired annotators

and closely monitored by the authors, discussing doubts and agreeing on the best course of action. The list of FAQs were gradually extended as more insight was obtained. For consistency, raters were asked to validate groups of related classes (e.g., sibling categories), so they could get familiar with specific sections of the ontology [66]. They were instructed to perform the task using high-quality headphones in a quiet environment, and taking periodic breaks to mitigate fatigue. During the campaign, the hired raters acquired solid expertise on the annotation task and a deep knowledge of the ontology. Therefore, we consider them experts for this task.

The outcome of the annotation campaign was 51,684 clips considered valid for the dataset, that is, with at least one "Present" label. All the "Present" labels amount to 59,981, all of them being the result of inter-annotator agreement, except 3390 which include labels with: *i)* only one PP rating and one PNP rating (and nothing else). This can be considered inter-annotator agreement at the "Present" level; *ii)* only one PP rating (and nothing else); *iii)* only one PNP rating (and nothing else). The two latter do not meet our definition of ground truth and could be more prone to errors, but were still considered to slightly increase the amount of data. It must be noted that the set of labels at this point comes from the validation of candidate labels proposed by a simple nomination system, which ultimately relies on the user-provided Freesound tags. Hence, it is to be expected that some sound events are not covered by the user-generated tags, or they are not proposed by the nomination system, leading to missing "Present" labels, a common phenomenon in large sound event datasets [42, 67]. That is, the resulting pool of audio clips have labels that are mostly correct (estimated at 94.3% in Sec. IV-C), albeit potentially incomplete which is problematic in evaluation. To address this issue, after splitting the data into development and evaluation sets (Sec. III-F), the latter is refined using another annotation tool (Sec. III-G). We define *correct* label as a label accompanying an audio clip that accurately identifies a corresponding sound source in the clip. We define *complete labels* for an audio clip as the set of labels that identify *all the target* sound sources in the clip as per a predefined vocabulary.

F. Data Split

The input to this stage is a pool of 51,684 audio clips with mostly correct labels (albeit potentially incomplete). The goal is to split the data into two subsets: *development* and *evaluation*. The development set will be used for training and validation. The evaluation set will be used for system benchmarking after exhaustive annotation. As stated in Sec. III-A, the evaluation set is our priority. A high quality evaluation set must be comprehensive, varied, and representative [68], while being free from contamination from the development set in order to allow testing models' generalization capabilities.

1) Split Criteria: We set four criteria for the split.

Non-divisibility of uploaders. The issue of *contamination* must be considered when splitting audio data, especially if portions of the data share a common pattern that brings acoustic similarity among its constituents. In Freesound, audio content is uploaded by users (in the following, *uploaders*). The uploaders can be very diverse: some are *small*—they upload a small number of audio clips (e.g., up to only few tens)—while other uploaders contribute with hundreds of clips. In the latter case, it can happen that some of the uploaded clips share the same sound source and/or physical location and/or recording gear (e.g., several notes of the same music instrument or vocalizations of the same pet). If some of these recordings are used for training and others for evaluation, their similarity may lead to overly optimistic performance, reflecting the classifier's ability to overfit development examples. As a result, this classifier may suffer from performance drop when tested on unseen data. This issue can be called *weak contamination* between development and evaluation, although, for simplicity, we will refer to it as *contamination* hereafter.[20] This phenomenon has been detected in computer vision benchmarks like CIFAR-10 and CIFAR-100 [44]. Another example of this in the field of music recognition is the denominated "album effect" [69, 70] or "artist effect" [71]. To avoid this issue, we make sure that *all* the content of each uploader is allocated either in the development or evaluation set. By doing this we assume that the evaluation performance reflects the model's ability to generalize to new audio material and recording conditions.

Small uploaders for evaluation. To obtain a varied evaluation set, it seems reasonable to allocate the content from small uploaders as it guarantees a higher diversity of sound sources, acoustic environments and recording equipment. In addition, a closer look at the Freesound data distribution revealed that recordings uploaded by small uploaders tend to

---

[20]This should not be confused with *data leakage*, which happens when the *same* (not similar) examples are used for both training and evaluation.

be slightly longer. It can therefore be expected that, in general, these longer recordings tend to contain more sound events when compared to shorter clips—a considerable portion of Freesound consists of short clips of few seconds featuring a single event. Under this assumption, longer recordings would be more real-world representative (see Sec. IV-B). Also, this is a more interesting content to further annotate exhaustively, and also with timestamps to allow future SED evaluations.

A coarse class distribution is enough. A fine-level split carefully matching a target class distribution is not needed at this point, as during the exhaustive labelling of the evaluation set we expect some classes to grow (Sec. III-G). This will create an imbalance that will need to be compensated.

Focus on leaf nodes. Among the classes available at this point, we focus on the subset of 113 leaf nodes with more than 100 clips as they are considered the most important classes.

2) Split Method: Given the many constraints, off-the-shelf methods such as random sampling, iterative stratification [72] or combinatorial optimization algorithms like knapsack problems [5, 73] are not well suited. Therefore, we implement an *ad hoc* approach consisting of iteratively allocating uploaders' content to the evaluation set after sorting them appropriately. First, we compute a score per uploader u as:

$$\text{score}^u = \text{n\_labels}^u_{\max} + \frac{1}{K_u} \sum_{k=1}^{K_u} \text{n\_labels}^u_{c_k}, \qquad (1)$$

where $\text{n\_labels}^u_{\max}$ is the maximum number of labels provided by the uploader u in any class, $\text{n\_labels}^u_{c_k}$ is the number of labels provided by u in the class c_k, and K_u is the number of classes *touched* by u (i.e., those to which u contributes). Uploaders are sorted in ascending score order and the content of low-score uploaders is transferred first. With the first term we prevent uploaders with abundant content concentrated in one specific class, and with the second term preference is given to users with low average number of labels per class for diversity. We found out that, by splitting the target 113 leaf nodes, some content corresponding to the remaining classes is automatically allocated due to the uploaders contributing also to them. This content is deemed sufficient as a fine-level class distribution is not the target at this point. We then proceed to allocate data to the evaluation set following the process shown in Algorithm 1. We traverse the $C = 113$ classes starting from

---

Algorithm 1: Data allocation to evaluation set

Data: Empty evaluation set per-class
  $E = \{e_{c_i} = 0\}_{i=1}^{C}$, uploaders ranking $\boldsymbol{u}$

1 **for** *class* $c_i \in C$ **do**
2   get current evaluation target t_{c_i}
3   **while** $e_{c_i} < t_{c_i}$ **do**
4   get next uploader u in ranking $\boldsymbol{u}$ with data in c_i
5        $e_{c_i} \leftarrow e_{c_i} +$ data from u in c_i
6   **for** *class* $c_k \in K_u$ **do**
7           $e_{c_k} \leftarrow e_{c_k} +$ data from u in c_k
8   **end**
9   **end**
10 **end**

Result: A candidate evaluation set

---

the least-represented ones since they have less flexibility for data allocation. For each class c_i, we progressively allocate content from the ranked uploaders $\boldsymbol{u}$ until a target amount of data t_{c_i} is reached. t_{c_i} is proportional to the total class label count, and rectified to lie in the range from 50 to 100 labels per class. By default, the maximum uploader size per class c_i (i.e., the maximum number of clips that one uploader u is allowed to contribute to c_i) is set to $0.1t_{c_i}$. Thanks to the proposed sorting, uploaders in the evaluation set do not reach such a maximum in the majority of classes (they often provide one single clip)—if they do, excess clips are discarded in most cases. However, due to the high uploader diversity, the maximum uploader size had to be increased in a few exceptions. Using the proposed scheme, we processed all the 7229 uploaders and we allocated 2794 of them to the evaluation set, totalling 11,466 clips.

The result is two pools of clips disjoint in terms of uploaders: a candidate development set and a candidate evaluation set. The latter is exhaustively labeled in the refinement task.

G. Refinement Task

As mentioned in Sec. III-E, in some clips, the current label sets could be an underrepresentation of the audio content, biased by the idiosyncrasies of the labeling pipeline. When evaluating classifiers' predictions against these incomplete label sets, classifiers can be penalized when predicting a correct label that happens to be missing from the ground truth. To address this issue, we refine the labels in the evaluation set. The goal is to obtain an *exhaustive* labelling, that is, a labelling as close as possible to the correct and complete transcription of the audio content (for the considered vocabulary of 395 classes).

1) Annotation Tool: We designed and implemented an annotation tool that allows two subtasks: *i)* to review the existing labels, and *ii)* to add missing "Present" labels. The subtask of adding missing "Present" labels has a considerable complexity since audio clips can sometimes contain very different sound events. Therefore, the success of this task relies on two key factors: *i)* raters with a deep understanding of the ontology, the agreed FAQs, and the particularities of the audio material; *ii)* an interface that facilitates exploration of the large vocabulary of the ontology. In regard to the first factor we turn to the pool of hired raters (4 of the initial 6), who acquired a solid expertise by extensive participation in the validation task (Sec. III-E). As for the second factor, the refinement task we implemented in FSA includes a tool to interactively explore different depth levels of the ontology (Fig. 4). This tool is based on a previous version described in [74]. A search input box allows to quickly navigate to classes in the table, where their hierarchical context is shown. For each class, textual descriptions and representative sound examples are displayed. The interface facilitates the comparison of different classes by simultaneously displaying their information.

2) Annotation Process: Clips were presented grouped by sound class to facilitate the task. For every class:

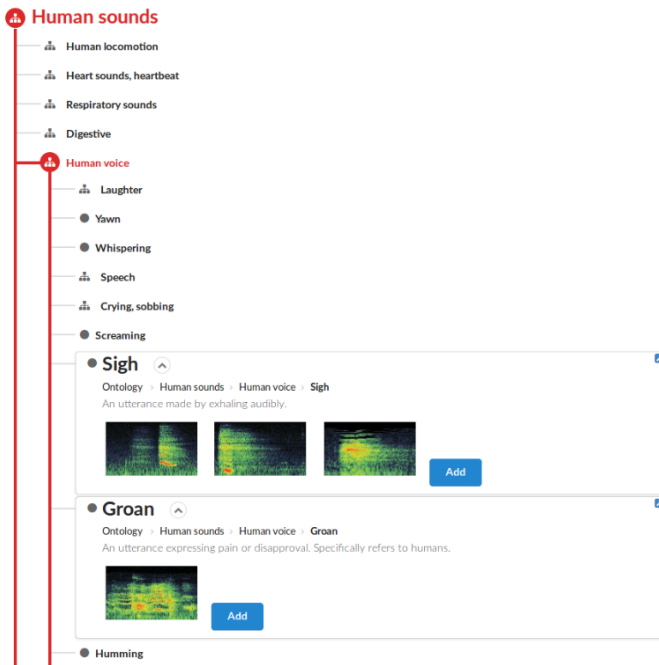1) raters were instructed to go through a **training phase** (same as in the validation task—see Fig. 2).

Fig. 4. Table for exploring the ontology in the refinement task.

2) For every clip, they would first **review existing labels** and modify them if needed. Then, they would **add any missing labels** by exploring the ontology (Fig. 4).

Raters were instructed to provide the most specific labels possible (typically leaf labels) as they are the most informative type of supervision. The quality assurance practices described for the validation task were also applied in the refinement task. Following this procedure, each evaluation label was verified or reviewed by between two and five different annotators (considering both validation and refinement tasks), including at least one expert. As a result, labels are expected to be correct and complete in the vast majority of cases (see Sec. IV-C). The exhaustive labelling carried out has two implications. First, absence of labels means absence of sound events (except human error)—a desired feature. Second, some classes are now much more represented than before as they are prevalent but were underrepresented, thus creating a class imbalance.

The outcome of this stage is a pool of exhaustively labeled clips (for the considered vocabulary), the majority of which will form the evaluation set.

H. Post-processing

This stage starts from two sets of data: a candidate development set with correct but potentially incomplete labels, and an exhaustively-labeled candidate evaluation set. The vocabulary used so far comprises 395 classes, yet many of them have little data (few tens of clips). While they may not be adequate for deep learning approaches, they can be useful for other practices, e.g., few shot learning [75]. Likewise, this information can provide insight as to the specific content of the dataset. Therefore, we provide two different formats for the annotations in FSD50K: *i)* The raw outcome of the annotation process, featuring all generated class labels without any restriction. These include classes with few data. We call this the *sound collection* format. *ii)* The outcome of curating the raw

annotations into a machine learning dataset with emphasis in sound event recognition tasks. This process involves, mainly, merging low prior classes into their parents thus ensuring a minimum amount of per-class data. This is what we define as *ground truth* for FSD50K, with a vocabulary of 200 classes. In Appendix B we describe the post-processing carried out to obtain what's finally released as FSD50K (consisting of a set of audio clips and corresponding ground truth). The post-processing stage includes determining FSD50K's vocabulary, balancing development/evaluation sets, creating a validation set, and propagating the labels hierarchically. Further technical details about the sound *collection* format can be found in FSD50K's Zenodo page.[4]

IV. DATASET DESCRIPTION

FSD50K is an open dataset of human-labeled sound events containing 51,197 clips unequally distributed in 200 classes drawn from the AudioSet Ontology. The dataset is freely available from Zenodo.[4] Hereafter, we refer to *development* (composed of *training* and *validation*) and *evaluation* sets described in the previous Sections as *dev*, *train*, *val*, and *eval*.

A. Characteristics

FSD50K is composed mainly of sound events produced by physical sound sources and production mechanisms. It also includes some classes that can inherently encompass several more specific sources (e.g., *Train*), some classes that do not relate to a specific source but to the perception of sound (e.g., *Clatter*), and few abstract classes (e.g., *Human group actions*). The dataset has 200 sound classes (144 leaf nodes and 56 intermediate nodes) hierarchically organized with a subset of the AudioSet Ontology [26]. The vocabulary can be inspected in Fig. 7. Note, however, that in some cases one leaf node in FSD50K (e.g., *Camera*) may be an intermediate node in AudioSet due to the fusion of low-occupancy classes (e.g., *Single-lens reflex camera*) with their parents. Following AudioSet Ontology's main families, the FSD50K vocabulary encompasses mainly *Human sounds*, *Sounds of things*, *Animal*, *Natural sounds* and *Music*. The vast majority of the content corresponds to sounds recorded from a sound field, while a small portion corresponds to sounds captured directly from electronic devices, typically in the context of musical instruments, e.g., some bass drums are generated with drum machines. The main characteristics of FSD50K in terms of number of clips, labels, duration and uploaders are listed in Table IV.

TABLE IV
MAIN STATISTICS FOR FSD50K

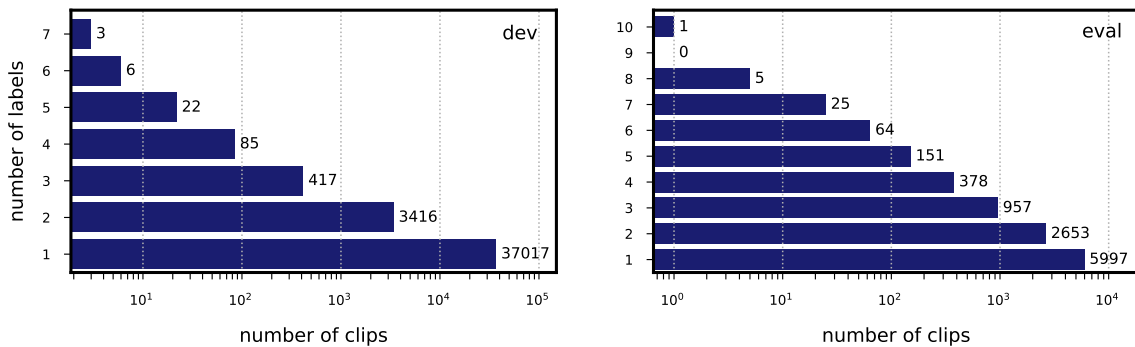|	total	dev	eval
clips	51,197	40,966 (80%)	10,231 (20%)
labels (unpropagated)	62,657	45,607 (72.8%)	17,050 (27.2%)
avg labels/clip	1.22	1.11	1.67
labels (propagated)	152,867	114,271	38,596
clips w/ leaf label(s)	40,461	31,310	9151
duration	108.3h	80.4h (74.2%)	27.9h (25.8%)
avg duration/clip	7.6s	7.1s	9.8s
uploaders	7225	4936	2289

Fig. 5. Label distributions in dev (left) and eval (right) sets. Clips in eval tend to have more labels (by dataset curation). Xaxis scale is logarithmic. Number of labels is reported in the *unpropagated* form. Note that visualization span differs among plots.
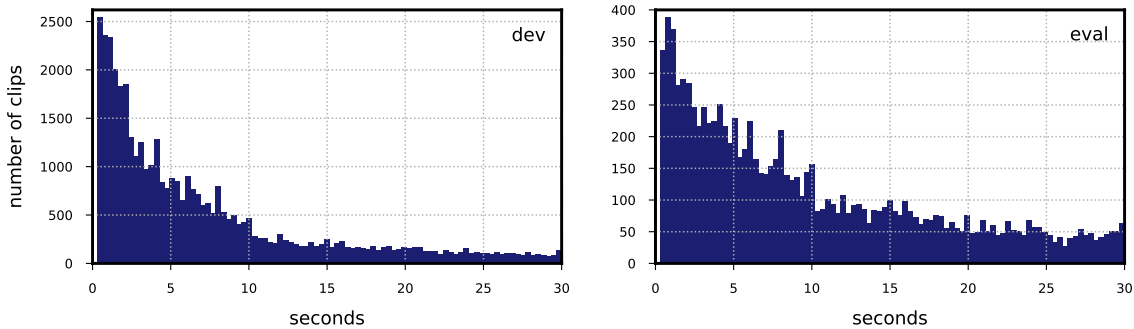

Fig. 6. Audio clip length distributions in dev (left) and eval (right) sets. Clips in eval tend to last slightly longer (by dataset design). Bins correspond to 1/3 second. Note that visualization span differ among plots.

The audio clips are grouped into a dev split and an eval split such that they do not have clips from the same uploader. Eval is exhaustively labeled, that is, annotations are correct and complete for the considered vocabulary (except human error). In dev, a small amount of content is exhaustively labeled, but the vast majority is composed of labels that are correct but could be potentially incomplete (see Sec. IV-C for label noise estimations). The number of labels is expressed in *unpropagated* and *propagated* forms. The number of *unpropagated* labels includes only the most specific labels per clip. It must be noted that this way of counting labels ignores a few labels in cases where a sound event co-occurs with: *i)* events from low prior siblings that were merged with their parent; *ii)* events that do not fit semantically in any other sibling provided by the ontology, hence they are annotated with their parent. While these cases are not frequent, the true number of human-provided labels describing sound events would be slightly larger than the one reported here. *Propagated* labels refer to the labels after hierarchical propagation (App. B). We use the unpropagated version to compute the average number of labels per clip. Note the increased number of labels per clip in eval due to the exhaustive labelling process, as can also be seen by comparing the label distributions in Fig. 5. In eval, all classes are present in both the singly-labeled data and the multi-labeled data forming it, except five classes that only appear in the multi-labeled data. In dev, all classes are present in both the singly-labeled data and the multi-labeled data forming it, except four classes that only appear in the singly-labeled data. A total of 31,310 clips are labeled with, at least, one leaf label in dev—the remaining 9656 clips are labeled only with intermediate node labels. This proportion

changes significantly in eval, where the majority of clips have leaf labels (9151 out of 10,231)—this is because in the refinement task raters were instructed to provide the most specific labels possible. All provided ground truth labels are hierarchically propagated, i.e., consistently propagated to their ancestors in the hierarchy. PP/PNP ratings are provided for the labels validated in the validation task. Out of the 108.3 hours of human-labeled audio, 31.5 are exhaustively labelled, most of them used for evaluation purposes (eval and val). The audio clips are of variable length ranging from 0.3 to 30s. Note the increased average duration of eval clips due to the allocation process (Sec. III-F), which can also be noticed by comparing the clip length distributions in Fig. 6. The ground truth labels are provided at the clip-level (i.e., weak labels). The dataset is sourced from 7225 Freesound users and the content was uploaded from Freesound's launch in 2005 until early 2019.

The number of clips per leaf class varies, roughly, from 40 to 200 in eval, and from 50 to 500 in dev, with a few exceptions. The number of clips in the intermediate nodes grows much more depending on the hierarchy. Therefore, class imbalance comes from two sources: non-uniform class distribution and variable-length of clips. The dataset is licensed under CC-BY license—nonetheless, each clip has its own specific license (CC0, CC-BY, CC-BY-NC or CC Sampling+, where CC0 and CC-BY amount to 84.7% of the dataset). When original audio clips have more than one channel, they are downmixed to mono. All clips are provided as uncompressed PCM 16 bit 44.1 kHz mono audio files. Further details about data licenses, ground truth format, and additional provided metadata can be found in the FSD50K Zenodo page.[4]

B. Discussion

1) Variable Clip Length and Weak Labels: Labels in FSD50K are provided at the clip-level (i.e., weak labels). However, unlike other sound event datasets featuring audio clips of same or similar lengths [26, 39, 76], FSD50K is composed of variable-length clips in the range [0.3, 30] seconds (Fig. 6). This provides FSD50K with a particular feature. On the one hand, some clips contain sound events where the acoustic signal fills almost the entirety of the file, which can be understood as strong labels. To give a sense of this, 12,357 clips in the dev set are shorter than 4s and bear one single label validated only with PP ratings. Thus, we estimate that the dev set is composed mainly of weakly labeled data and a portion of strongly labeled data, in a rough proportion of 70%/30%. On the other hand, another small portion of the data presents a much weaker supervision—e.g., 9494 dev clips are longer than 10s (Fig. 6). The longer the clips, the lesser the certainty of where the labeled event is actually happening. This is referred to as *label density noise* [77], defined as a measure of the weakness of labels for a given weakly labeled clip. The impact and limitations of weak labels in sound event recognition (SER) are discussed in [32, 78]. Audio processing can be done by handling the variable-length clips *as is*, or by slicing the clips into fixed-length patches. Utilizing variable-length inputs in the context of deep networks entails certain architectural constraints, such as using fully convolutional networks or appropriate pooling strategies. To avoid these constraints, variable-length clips are sometimes split into fixed-length patches, which can be processed in several ways. A simple way consists of inheriting the clip-level labels by every constituent patch (a practice called *false strong labeling* in [79]), which can generate false positives if the label is not active in a given patch. In this case, for evaluation, patch-level scores must be aggregated into clip-level predictions to be compared against the weak labels. Another alternative to process fixed-length patches belonging to weakly-labelled clips is to adopt a multiple-instance learning strategy [43].

2) Audio Quality: Given the heterogeneity of Freesound audio it is difficult to make strong objective claims about audio quality in FSD50K. Nonetheless, upon inspection of the clips' metadata, it can be seen that many Freesound users utilize (semi-) professional recording equipment (e.g., microphones or preamplifiers of brands such as *Neumann*, *Rode* or *Tascam*). Our experience after annotating the dataset is that the audio generally has relatively high SNR and dynamic range. To put this into context, we note that the notion of audio quality in sound recognition datasets has changed over time. In early DCASE Challenges, datasets recorded with professional equipment dominated, some of them being recorded with one single microphone model [13, 23, 80, 81]. Then, AudioSet became popular, in which a large variety of devices are used for recording YouTube *videos* (where audio quality is not necessarily a priority), and often including lower SNR conditions. We extracted the global SNR for FSD50K and AudioSet (its eval set and balanced train set) using the ITU-T P.563 [82]. It must be noted that P.563 is designed for evaluating human speech, while the content of both datasets is much more diverse. Therefore, strong claims cannot be made based on this measurement. We use it as a common-referenced, rough indication of SNR given that it is not trivial to accurately compute SNR for the different types of audio under consideration. SNR values reported in Table V are mean and median of per-clip SNR values. The mean SNR for FSD50K is greater than that of AudioSet. For AudioSet, it can also be seen that the mean SNR is greater than the median, suggesting that the SNR distribution is positively skewed, i.e, lower SNR values are more frequent.

3) Real-world Audio: Many clips in Freesound are real-world recordings of sound events happening in the wild, e.g., a car passing by. However, it is not uncommon that some sound events are recorded under careful conditions in order to obtain clean and isolated high-quality sounds, as in a foley sound setting (e.g., the sound of tearing paper carefully located in front of a microphone). Further, a few clips in Freesound consist of sound events purposely generated with the sole objective of being recorded, e.g. a faked laughter. While these recordings are valuable for sound design, in some cases they could feature a lack of naturalness or acoustic mismatch with respect to sound events in the wild. This may question the suitability of a portion of the data for learning sound recognizers to be deployed in the wild, where more adverse generation and recording conditions can be encountered. To what extent this affects models' generalization to adverse scenarios is an open question. Mitigating this potential issue could be a research problem involving, for example, data augmentation [83] or domain adaptation [84] techniques.

C. Limitations

1) Label Noise: Throughout this paper we have discussed the correctness/completeness of labels in the dataset. While we aimed at full label correctness and completeness, this is unrealistic as it would mean perfect accuracy of the nomination system that proposes candidate labels (Sec. III-D), and of the human-provided labels. As supervised learning research moves towards larger datasets, issues of label noise become inevitable. For instance, labeling error in AudioSet is estimated at above 50% for $\approx$18% of the classes.[10] Similarly, ImageNet data are often presumed to have correct labels, but it has been estimated that at least 100k images could be labeled incorrectly [85]. In SER, label sets in not-small datasets are inherently noisy due to reasons like sub-optimality of automatic methods used in the creation, or the difficulty of annotating audio—especially without visual cues, with large vocabularies, and because the annotation process is, sometimes, inherently subjective and ambiguous.

Despite our efforts to mitigate label noise in FSD50K, there are still a few label noise problems. The main problem is the existence of *missing* "Present" labels (false negatives). These are labels that would be included in an ideal exhaustive annotation but which are missing from the current set. Recent work identifies this as a pathology in AudioSet as well, and proposes a method to tackle it [42]. This problem affects the dev set more due to the annotation process based on validation of previously nominated labels—if sound events are not nominated by the system, they lack labels (Sec. III-E).

This may happen with sound events that tend to be less represented by the Freesound user-provided tags, such as human or bird sounds when they are not the most relevant events in a clip. Because the eval set received exhaustive annotation, this problem is minimized there. To a much lesser extent, two additional sources of missing labels exist. First, the impossibility of propagating labels in the hierarchy when multiple ambiguous paths are encountered—again, this affects more the dev set (see App. B). Second, missing labels can occur as a result of annotating with a finite vocabulary— there may be additional acoustic content *out-of-vocabulary*. In particular, in the annotation process, when we encounter a sound event out-of-vocabulary, two cases can happen: *i)* if the event is alone in the clip (or with other out-of-vocabulary events), the clip is discarded; *ii)* if the event co-occurs with other events *in-vocabulary*, then the clip is kept, but the out-of-vocabulary event remains unlabeled. Apart from missing labels, the other label noise problem is *incorrect* "Present" labels (a false positive, and potentially a false negative if the true class is in-vocabulary). This is the result of human annotation errors. Because we adopted mechanisms to bootstrap human annotation quality (Secs. III-E and III-G), we expect incorrect labels to be rare (see below). Both missing and incorrect labels would be class-conditional as some classes are clearly more ambiguous than others. More details about label noise characterization can be found in [32].

We can use the refinement task processing to quantify the label noise at the output of the validation task. A total of 11,847 audio clips (carrying a total of 13,681 labels) were processed with the refinement task, the majority of which ended up in the eval set. The processing undergone by these clips in this task in order to approach a complete sound event transcript is summarized next, in order to quantify the label noise at this point. First, we estimate the amount of missing labels. A total of 6030 clips (50.9%) received at least one additional label, indicating that there was some unlabeled material. For these 6030 clips, a total of 10,473 labels were generated. Second, we estimate the amount of incorrect labels. The incoming 11,847 audio clips before the refinement task featured a total of 13,681 labels, out of which 773 (5.7%) were then rejected by the annotators. This means that 94.3% of the incoming labels were verified as correct. This gives a sense of the level of label correctness obtained with the validation task (and therefore, the dev set). In sum, after the validation task, for a pool of 11,847 audio clips mostly selected for eval (carrying a total of 13,681 labels), 94.3% of the incoming labels were verified as correct, and 50.9% of the incoming clips had one or more missing labels. This indicates that in the dev set the labels are mostly correct, yet there is a large amount of clips featuring missing labels. However, the actual number of clips with missing labels in the dev set is estimated as less than 50.9% because, as explained earlier, clips selected for dev tend to have less average number of sources per clip than those selected for eval. For the eval set, we have not quantified the amount of correctness and completeness due to lack of resources. However, the amount of correctness is expected to be not very different from the 94.3% estimated at the output of the validation task. We expect this because

the hired raters are more qualified at the refinement task than during the previous validation task, as they have gained more annotation experience and a deeper knowledge of the ontology. Labelling errors in FSD50K can be reported via its companion site.[6] In this way, future dataset releases can include fixes reported in a collaborative way.

2) Data Imbalance: While some classes are abundant, others are much less represented due to the data scarcity in Freesound and/or low performance of the nomination system. Another source of imbalance is the variable length of clips— some classes tend to contain shorter/longer clips depending on the sound events and the preferences of Freesound users when recording them. Finally, the hierarchy of the ontology favours data imbalance between classes at different levels.

3) Data Bias in Development Set: Because we prioritized the allocation of small uploaders in the eval set to increase its diversity (Sec. III-F), the development portion of a few classes is dominated by a few large uploaders. Under the assumption that this signifies similar training examples in certain cases, this could create a data bias, which could be learned by models [86]. This happens mainly in a few musical instruments, e.g., *Trumpet*, due to the fact that Freesound users tend to upload many clips for these classes. Further analysis would be needed to determine if and how much this potential bias causes lack of generalization for these classes.

4) Lack of Specificity in the Vocabulary: Some leaf nodes in the ontology were merged to their parents due to data scarcity. For instance, leaf nodes such as *Blender*, *Chopping (food)*, and *Toothbrush* had to be merged with their parent *Domestic sounds, home sounds*. This motivated us to keep the latter class as a valid class despite that it is blocked in the ontology [26].

D. Applications

FSD50K allows evaluation of approaches for a variety of sound recognition tasks. The most evident is multilabel sound event classification with large vocabulary [33, 87]. In this context, the proposed dataset supports several approaches such as learning sound event representations directly from waveforms [88, 89]; analysis of label noise mitigation methods leveraging the non-exhaustive labeling of the dev set [32, 42, 90]; multimodal approaches using audio and text information (e.g., using the provided Freesound tags, title, and textual description for the clips) [91, 92]; evaluation of hierarchical classification via ontology-aware learning frameworks [5, 93, 94]; or approaches specifically combining strong and weak labels [95]. By leveraging the common vocabulary between FSD50K and AudioSet, we hope that a number of tasks become possible, such as experimenting with domain adaptation techniques [96], or cross-dataset evaluation [97] under different acoustic conditions. Other tasks include search result clustering in large vocabulary datasets [98] or universal sound separation [99]. In addition, FSD50K has already accomplished several milestones. A subset of the data has been used for a number of smaller datasets for classification [32, 33, 58, 100] and source separation [63]. Likewise, subsets of FSD50K have enabled several sound recognition Challenges—specifically, DCASE 2018 Task 2 "General-purpose tagging of Freesound audio with AudioSet labels" [58], DCASE 2019 Task 2 "Audio

tagging with noisy labels and minimal supervision" [33] and Task 4 "Sound event detection in domestic environments with weakly labeled data and soundscape synthesis" [28], and DCASE 2020 Task 4 "Improving sound event detection in domestic environments using sound separation" [101].

E. FSD50K and AudioSet

As FSD50K and AudioSet are based on the same ontology and thus are partially compatible, we discuss the main similarities and differences between both. Table V summarizes some of them. Both datasets use the AudioSet Ontology for

TABLE V
COMPARISON OF SOME PROPERTIES OF FSD50K AND AUDIOSET

	FSD50K	AudioSet
classes	200	527
content	waveform	features
dev clips	40,966	≈2M
eval clips	10,231	20,383
clip length	0.3-30s	≈10s
dev labeling	CpI	CpI
eval labeling	exhaustive	CpI
source	Freesound audio	Youtube video
train/val split	✓	-
P.563 SNR (mean, median) [dB]	(26, 25)	(14, 10)

organization, however FSD50K uses a smaller subset. All classes in FSD50K are represented in AudioSet, except *Crash cymbal* as well as four classes that are blocked in AudioSet but not in FSD50K (*Human group actions*, *Human voice*, *Respiratory sounds*, and *Domestic sounds, home sounds*). The official AudioSet release consists of audio features pre-computed at 960ms-resolution, released under CC-BY-4.0 license. FSD50K provides audio waveforms under several CC licenses as decided by Freesound users. In terms of stability, FSD50K is downloadable as several zip files from its Zenodo page.[4] AudioSet features can be downloaded as a tar.gz file from the AudioSet website.[2] The original YouTube video soundtracks are gradually disappearing as they are subject to deletions and other issues, and their usage may be affected by copyright policies (Sec. I). AudioSet's dev set is significantly larger than FSD50K's whereas AudioSet's eval set is roughly twice that of FSD50K. Since AudioSet has a vocabulary 2.6 times larger, this means that in some classes there is more evaluation content in FSD50K. Clips in AudioSet last ≈10s, whereas in FSD50K their length varies from 0.3 to 30s. Thus, label weakness is more homogeneous in AudioSet, whereas it varies significantly in FSD50K, yielding quasi-strong labels as clips get shorter, and much weaker labels in the longest clips.

In terms of labeling, FSD50K provides event predominance annotations ("Present and predominant" & "Present but not predominant", Sec. III-E) while AudioSet only provides presence annotations ("Present"). While it is not easy to objectively compare label quality in both datasets, we speculate that the labeling of both dev sets could be generally regarded as *Correct but Potentially Incomplete* (CpI), i.e., both dev sets would be affected by a certain amount of missing labels. However, it seems reasonable to assume that, in the FSD50K portion of rather short sounds with PP annotations (see Sec. IV-B), the amount of missing labels is minimal. This assumption is consistent with our observations from the annotation experience.

The eval set of FSD50K was exhaustively annotated; therefore, absence of labels means absence of sound events (except human error). By contrast, the eval annotations in AudioSet would be in general CpI, similar to those of the AudioSet train set. Unlike AudioSet, FSD50K consistently provides all relevant labels in a hierarchical path, except in a few specific cases of ambiguous ancestors. As additional resources, we provide additional metadata (e.g., Freesound tags and class-wise annotation FAQs) and allow flagging labeling errors.[6]

Finally, despite both datasets being highly heterogeneous, we make the following conjectures. Freesound clips are typically recorded with the goal of capturing audio, which is not necessarily the case in YouTube videos. Additionally, given the AudioSet size, its audio clips are presumably recorded with a higher diversity of devices. This would provide AudioSet with a higher diversity of audio qualities, often including more real-world and lower SNR conditions than Freesound audio (see Table V for a rough SNR estimation described in Sec. IV-B). Thus, a certain acoustic mismatch between both datasets may be expected. In our view, both datasets suppose complementary resources for sound event research.

V. EXPERIMENTS

In this Section, we conduct a set of multi-label sound event tagging (SET) experiments to give a sense of the performance that can be achieved with FSD50K using a baseline pipeline (Sec. V-B), and to learn about the main challenges to consider when splitting Freesound audio for sound event recognition (SER) tasks (Sec. V-C). For reproducibility, implementation details of evaluation metrics, learning pipeline, and networks can be inspected in the open-source code.[5]

A. Evaluation

Some common evaluation metrics for SET (e.g., F-score or overall error ratio) depend on an operating point, i.e., a decision threshold applied on the per-class output scores. These metrics encompass evaluation of the model's performance *and* of the decision threshold tuning. However, we believe that decoupling these two factors is desirable as, strictly, they are two different issues and the optimality of the latter can be application-dependent. Thus, we propose metrics able to evaluate a model's performance globally, integrating all possible operating points such that setting a decision threshold is not needed. This trend has been adopted in other fields such as speaker recognition [102] and also recently in sound event detection (SED) [103].

On the one hand, we use common within-class metrics, i.e., metrics that rank all test samples according to the classifier score for one given class. These metrics deal with only one classifier output at a time, such that calibration across different classifier outputs is irrelevant. Following [26, 104], we use mean Average Precision (mAP) and d'. mAP is the mean across classes of the Average Precision (AP), which summarises the precision-recall (PR) curve as the classifier decision threshold is varied. AP is calculated as the Precision (i.e., the proportion of positive samples in a ranked list) averaged across all the lists just long enough to recall a new

positive sample [104, 105]. AP is very similar to the area under precision-recall curve (PR-AUC), both being the most common ways of summarising a PR curve—the difference between them lies in implementation details [106, 107]. d' (d-prime) measures the separation between the means of two unit-variance normal distributions (corresponding to the scores for positive and negative examples) that would achieve the same ROC-AUC [108]. d' is computed as a monotonic transform of ROC-AUC [104, 108]:

$$d' = \sqrt{2}F^{-1}(ROCAUC), \qquad (2)$$

where F^{-1} is the inverse cumulative distribution function for a unit Gaussian. To complement the within-class metrics, we propose to use a between-class metric, i.e., which evaluates the overall ranking across all classifier outputs for every test sample. Specifically, we use *Label-weighted label-ranking average precision* (lωlrap), recently introduced for DCASE2019 Task 2 [33]. Let $C(s)$ be the set of reference labels for test sample s, $Prec(s, c)$ be the label-ranking precision for the list of labels up to class c, and $Rank(s, c)$ be the rank of class label c in that list. $Prec(s, c)$ is equal to 1 if all the top-ranked labels down to c are part of $C(s)$, and at worst case equals $1/Rank(s, c)$ if none of the higher-ranked labels are correct. Then, lωlrap can be expressed by

$$l\omega lrap = \frac{1}{\sum_s |C(s)|} \sum_s \sum_{c \in C(s)} Prec(s, c) \qquad (3)$$

where $|C(s)|$ is the number of reference labels for sample s. A Python implementation of lωlrap is provided online.[21] For all metrics, larger is better. mAP $\in [0, 1]$, non-pathological $d' \in [0, \infty)$, and lωlrap $\in [0, 1]$. All metrics are computed on a per-class basis, then averaged with equal weight across all classes to yield the overall performance (i.e., *balanced* a.k.a. *macro* averaging), as in [26, 42, 104].

B. Baseline Systems

Next, we benchmark several commonly used deep networks on the proposed FSD50K.

1) Learning Pipeline: Incoming audio is downsampled to 22.050 kHz and transformed to 96-band, log-mel spectrogram as input representation. To deal with the variable-length clips, we use time-frequency (T-F) patches of 1s (equivalent to 101 frames of 30ms with 10ms hop)—thus the input to all models is of shape TxF=101x96. Clips shorter than 1s are concatenated until such length, while longer clips are sliced in T-F patches with 50% overlap inheriting the clip-level label (a.k.a. false strong labeling [79]). We adopt the train/val split designed in App. B. We implement a learning pipeline in TensorFlow [109]. Models are trained using Adam optimizer [110] to minimize binary cross-entropy loss, with initial learning rate depending on the network (see Table VI), which is halved whenever the validation PR-AUC plateaus for 5 epochs (no tolerance). Models are trained up to 100 epochs, earlystopping the training whenever the validation

---

[21] https://colab.research.google.com/drive/1AgPdhSp7ttY18O3fEoHOQKlt_3HJDLi8

PR-AUC is not improved in 10 epochs. We use a batch size of 64 and shuffle training examples between epochs. Once the training is over, the model checkpoint with best validation PR-AUC is selected to predict scores and evaluate performance on the eval set. We optimize PR-AUC (instead of other metrics based on ROC curves) because PR curves can be more informative of performance when dealing with imbalanced datasets [111]. Likewise, we use PR-AUC (instead of mAP) for simplicity as it is a built-in metric in TensorFlow. For inference, we pass each (eval or val) T-F patch through the model to compute output scores, which are then averaged per-class across all patches in a clip to obtain clip-level predictions, as in [26, 42]. We note this aggregation must be done also for validation—preliminary experiments validating at patch-level using inherited clip-level labels revealed misleading results. Extensive hyper-parameter tuning (beyond learning rate) is not conducted.

2) Network Architectures: Current trends in SER encompass mainly CNNs [87, 104, 112]–[114] and CRNNs [20, 115]. We run experiments with the following networks, all of them ending with a fully connected layer of 200 units (the vocabulary size) with sigmoid activation to support multi-label classification. Main parameters for CRNN and VGG-like architectures are set via non-exhaustive preliminary experiments.

CRNN. This is one of the most used architectures for SED [20], and to a lesser extent for SET [116]. Our model, inspired by [20], has three convolutional layers of 128 filters with a receptive field of (5,5), each of them followed by Batch Normalization (BN) [117], ReLU activation and max-pooling. The max-pooling sizes are $(t, f) = (2, 5), (2, 4)$ and $(2, 2)$—since we are not interested in detecting events' timestamps, we pool also in the time dimension which reduces dimensionality without harming performance in our experiments. To model events' temporal structure in the incoming feature maps, the convolutional stack is followed by a bidirectional GRU layer of 64 units, returning the last output of the output sequence.

VGG-like. VGG-based architectures have been widely used for both SET [87, 118] and SED [119]. We use a model inspired by the original architecture [120] from computer vision, but reduced to a much smaller size. In particular, this model has three convolutional layers of 32 filters, two convolutional layers of 64 filters, and one convolutional layer of 128 filters. All convolutional layers have a receptive field of (3,3) and are followed by BN and ReLU activation. Between each group of convolutional layers with same number of filters, max-pooling of size (2,2) is applied. Output feature maps are summarized by concatenating global max pooling and global average pooling per channel. Summarizing the learnt audio representation via combination of these two poolings provided a small mAP boost with respect to using either of them individually. Then, the outcome is passed through a fully connected layer of 256 units.

Finally, we also experiment with two architectures taken off-the-shelf from the computer vision literature. While the two previous networks received some tuning in their design, the next ones are the original architectures without any tuning whatsoever—only the input/output shapes to match our task.

TABLE VI
EVALUATION PERFORMANCE FOR THE ARCHITECTURES CONSIDERED

Model	lr	weights	mAP	d'	$l\omega lrap$
CRNN	5e-4	0.96M	0.417 ± 0.003	2.068 ± 0.015	$\mathbf{0.519} \pm 0.002$
VGG-like	3e-4	0.27M	$\mathbf{0.434} \pm 0.002$	$\mathbf{2.167} \pm 0.011$	0.514 ± 0.003
ResNet-18	1e-5	11.3M	0.373 ± 0.001	1.883 ± 0.020	0.465 ± 0.001
DenseNet-121	5e-5	12.5M	0.425 ± 0.002	2.112 ± 0.032	0.505 ± 0.004

ResNet-18. ResNets [121] have been sucessfully used for SER [42, 51, 112].

DenseNet-121. DenseNets are reported to outperform ResNets for image recognition [122], and have been recently used for SET [90, 123].

3) Results: Table VI lists the results for the considered architectures, along with the learning rates used (after tuning) and the number of weights. Each network is trained from scratch three times with different random initialisation and different orderings in the training data. We report average and standard deviation of the evaluation performance across the three trials. The following results and discussion are based on the particular train/val/eval split used. Interestingly, the best overall model across all metrics is VGG-like, despite being less modern and more lightweight than the other architectures. This result accords with similar recent findings in music genre recognition [124]. The building blocks of VGG-like are very similar to those of the *CNN14* network in [112], which rivals state-of-the-art results in AudioSet classification [112]. However, CNN14 is much deeper and heavier (81M weights). The VGG-like model is closely followed by DenseNet-121, which has many more weights, and then by the CRNN, which shows the best *lωlrap*. CRNN architectures are also used in some top SED systems, e.g., in recent DCASE Challenge Task 4 editions [28, 101]. ResNet-18 is found to be the worst performing model. Curiously, we also observe that the optimal learning rates tend to be rather low for this architecture. We also tried ResNet-34 in preliminary experiments, obtaining similar results (at the expense of many more weights). Our results contrast with the successful results of [112, 125], which achieved state-of-the-art performance for AudioSet classification using ResNet architectures. Factors possibly influencing this different behaviour include the different amount of training data (much larger in AudioSet) or the data itself. Results in Table VI suggest that, for our scale of data, smaller models with basic tuning and audio-informed design choices can outperform much larger off-the-shelf computer vision architectures; however, DenseNet-121 with no tuning provides good performance.

Fig. 7 shows the per-class AP (averaged across three trials) for all classes in FSD50K, using the best-performing VGG-like model (dark blue), and the CRNN model (light blue). Leaf nodes with top recognition include *Applause*, *Burping, eructation*, *Purr*, and *Computer keyboard*, with AP over 0.75. The worst performance is shown in *Boat, Water vehicle*, *Cowbell*, *Speech synthesizer*, *Tap* and *Tick*. After inspection of the latter classes, we conjecture this is due to aspects such as high intra-class variation, confound with other similar classes, ambiguity in the class definitions, or very short length of sound events—all of them being relevant challenges in SER. Finally, it can be seen that most per-class APs by the CRNN are

slightly lower than those of the VGG-like model—as expected since VGG-like has a higher overall mAP. However, there are a few exceptions in which the CRNN performs better, such as in different types of speech (either spoken, sung, screamed, yelled or whispered). This is interesting as CRNNs were originally proposed for speech recognition [126] before being adapted for SER [20]. Other exceptions include some human sounds (e.g., types of laughter (*Chuckle, chortle* or *Giggle*), *Gasp*, or *Crying, sobbing*), as well as some animal vocalizations (e.g., *Bark*, *Meow* or *Chicken, rooster*). These sounds share a marked temporal behaviour. These results highlight the different behaviour, for some classes, of a model including a recurrent layer with respect to another relying only on convolutional layers.

C. Impact of Train/Validation Separation

In App. B we discuss some factors to consider when splitting Freesound audio data for machine learning, and we design a validation set emphasizing the issue of data contamination. Here, we experimentally analyze the impact of contamination in this setting. To this end, we pick one architecture from Sec. V-B (the CRNN) and we train and evaluate it utilizing three different train/validation splits in order to show their differences. Specifically, let us consider three candidate validation sets obtained with different approaches:

1) **val_random** is computed via *random* sampling. We run 3000 trials of a train/validation separation and we select the validation set with minimum Jensen-Shannon (JS) divergence[22] with respect to the development set.
2) **val_is** is computed via *iterative stratification* [72]. We run 3000 trials of a train/validation separation and we select the validation set with the minimum number of shared uploaders between training and validation.[23]
3) **val** is the validation set proposed in App. B.

In all cases, the validation set is initialized with most of the data that was transferred from the first evaluation set prototype to the development set for balancing purposes (App. B). Since this content is exhaustively labeled, it is well suited for evaluation purposes. The main characteristics of the three validation sets are listed in Table VII. The sets *val_random* and *val_is* amount to ≈15% of the development data associated with leaf nodes (Table IV); *val* is slightly lower (13.3%) due to allocating less validation data for the most abundant classes as well as some approximations (App. B). All validation sets have a similar duration. In terms of stratification, the split done through iterative stratification, *val_is*, yields more similar class

---

[22]The JS divergence is based on the Kullback-Leibler divergence but it is symmetric. We use it as a distance metric to measure similarity between the development and validation distributions, similarly as in [5].

[23]Minimizing the JS divergence is not needed here as stratification is already the objective of this method, hence all separations have a fairly consistent JS divergence.
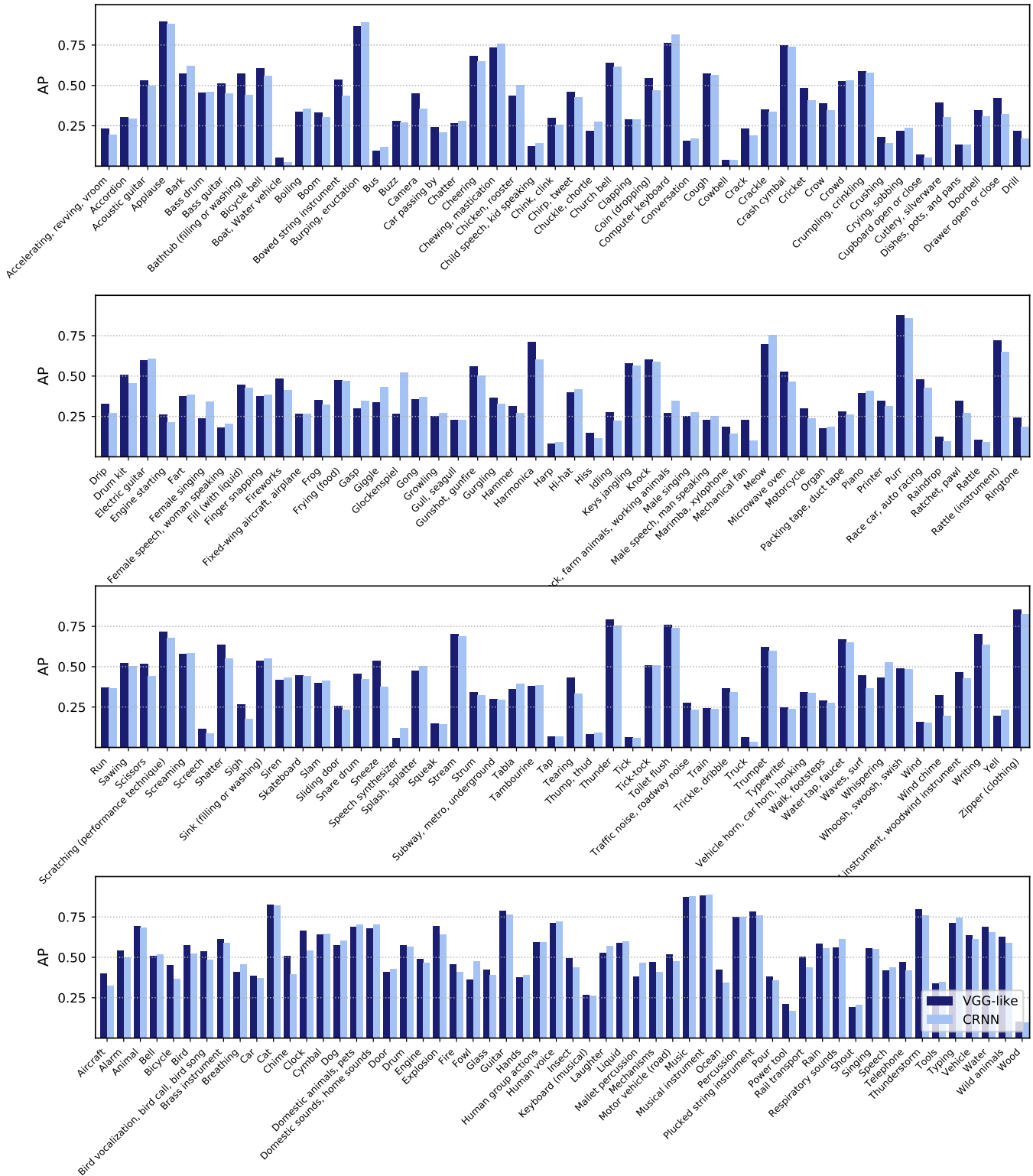
Fig. 7. Per-class average precision for all classes in FSD50K, using the best-performing VGG-like model (dark blue) and the CRNN model (light blue). Top 3 rows show the 144 leaf nodes and bottom row comprises the 56 intermediate nodes.

TABLE VII
MAIN STATISTICS FOR THE CONSIDERED VALIDATION SETS

Validation Set	clips	duration	JS	shared uploaders	PR-AUC drop
val_random	4697	9.7h	1.8×10^{-2}	930	0.15
val_is	4543	9.3h	6.8×10^{-3}	857	0.14
val (proposed)	4170	9.9h	2.1×10^{-2}	641	≈ 0

distributions than the other two, which are on par. The main differences lie in the uploaders "shared" between train and validation, both in number and in their nature. In particular, *val_random* and *val_is* suffer from within-class (WC) and between-class (BC) contamination as no measure is taken to prevent them. By contrast, *val* is designed to minimize WC contamination while being relatively flexible with BC contamination (see App. B for definitions of WC and BC contamination, as well as for the design of *val*). Therefore, not only is the number of shared uploaders less in the proposed *val*, but also the contamination is limited mostly to BC.

To compare the candidate splits, we train the CRNN of the previous Section using the three of them (in this case, with a learning rate of 1e-4 and no learning rate scheduling). Fig. 8 illustrates the learning curves (PR-AUC for train, validation, and evaluation) using each of the splits. We display 60 training epochs allowing validation and evaluation performance to roughly stabilise. From Fig. 8 and Table VII several observations can be made. In the left and middle plots of Fig. 8 (corresponding to *val_random* and *val_is*), validation performance is substantially better than evaluation performance. In these cases, the classifier is trained and validated on clips from the same uploader *and* the same class (i.e., WC contamination). We call this the "uploader effect" (following the analogy of the "album effect" [70] or "artist effect" [71]). In Table VII, it can be seen that the number of uploaders shared between train and validation is positively correlated with the validation-evaluation PR-AUC drop. In the cases of *val_random* and *val_is* we observe substantial performance drops, whereas with *val* the performance drop is very small. Results from Fig. 8 and Table VII suggest that, when contamination is considered and minimized, validation performance is a good proxy of evaluation performance—otherwise, it can be overly optimistic. Consequently, if the model is tuned using the validation set it may occur that, depending on the type and amount of contamination, the tuning reflects model's ability to partially overfit train data rather than to generalise to unseen data. In addition, our results indicate that the distinction between WC and BC contamination seems reasonable in the context of Freesound audio organized with a large vocabulary, confirming our initial hypothesis that WC is the most harmful type while BC has lesser impact (see App. B).

Lastly, we observe a slightly higher train performance and slightly lower validation and eval performances when using *val* (right plot of Fig. 8), which content comes mostly from a variety of small uploaders. Under the assumption that not all training examples are equally informative (which is the basis for disciplines like instance selection [127]), this may occur because the content transferred to *val* includes some highly informative examples. Yet, we propose this train/validation split

for systems' benchmarking because we deem it more methodologically correct than the others given that data contamination is minimized. In summary, carefully splitting Freesound audio is important as it can have a non-negligible impact on learning and performance. Therefore, for reproducibility and fair comparability of results, system benchmarking should be done explicitly specifying the validation split that was used.

VI. SUMMARY AND CONCLUSION

In this paper, we introduced *FSD50K*, a dataset containing 51,197 Freesound clips totalling over 100h of audio manually labeled using 200 classes drawn from the AudioSet Ontology. The audio clips are CC-licensed, thereby making the dataset freely distributable (including audio waveforms). We proposed a methodology for creating datasets of sound events based on human validation and refinement, and using a mixture of crowd-sourcing and recruited trained annotators. In this process, we experienced how human labeling of a large vocabulary of everyday sounds is a laborious and complex task. Special emphasis was put on the careful curation of the evaluation set content and labels, so that it can serve as a reliable evaluation benchmark. We showed how it is important to adapt the dataset creation process to the specifics of the source data—in our case, Freesound audio and metadata, and the AudioSet Ontology—and how a deep knowledge of these data is crucial to identify data challenges and limitations, and to avoid pitfalls in the creation of the dataset. Finally, through experimental results we showed that, for FSD50K classification, smaller models with basic tuning and audio-informed design choices can outperform larger off-the-shelf computer vision architectures. We also showed that within-class data contamination must be considered when splitting Freesound audio as it can have a considerable effect on the evaluation of sound event classifiers. FSD50K is an open and stable dataset aimed at complementing AudioSet in order to foster reproducible large-vocabulary SER research. In the future, dataset extensions could be carried out. More data could be added via semi-automatic methods by leveraging models trained on FSD50K to scale up efficiently. Likewise, the vocabulary could be extended by growing the merged leaf nodes in FSD50K.

APPENDIX A
ONTOLOGICAL NOMENCLATURE

We clarify next some basic (albeit relevant) ontology-related terms used in this paper. We shall refer to the 632 classes in the ontology as *nodes* (either *leaf* nodes when they are located at the very bottom of the hierarchy, or *intermediate* nodes otherwise). We shall also use the ontological terms *children* and *parents*, as widely used in ontology-related genome research [128]. Note that, by definition, leaf nodes do not have children nodes, while the intermediate nodes do. Similarly, given a node, we refer to all the parent nodes connecting it to the root of the ontology as *ancestors*. As an example, let us consider the hierarchical path: *Root → Natural sounds → Thunderstorm → Thunder*. In this path, *Thunder* is a leaf node; *Natural sounds* and *Thunderstorm* are both intermediate nodes; *Thunderstorm* is child of *Natural sounds* and parent of
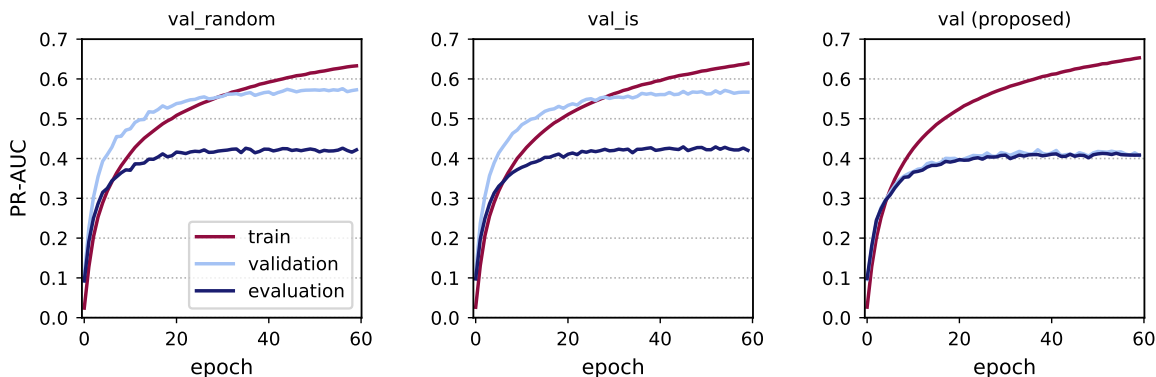
Fig. 8. Learning curves (PR-AUC for train, validation, and evaluation) for the CRNN model using the three train/validation splits specified in Table VII (*val_random* (left), *val_is* (middle), and the proposed *val* (right)). Validation performance is substantially better than evaluation performance when using *val_random* and *val_is*, in which the classifier is trained and validated on clips from the same uploader *and* the same class (WC contamination). When this type of contamination is minimized (*val*), validation performance is a good proxy of evaluation performance.

Thunder; and *Thunderstorm* and *Natural sounds* are all the ancestors of *Thunder*.

APPENDIX B
POST-PROCESSING STAGE FOR FSD50K

The outcome of the refinement task (Sec. III-G) is two sets of data: a candidate development set with correct but potentially incomplete labels, and an exhaustively-labeled candidate evaluation set, using a vocabulary of 395 classes. Here, we describe the post-processing carried out to obtain what's finally released as FSD50K, using a vocabulary of 200 classes.

A. Determine FSD50K vocabulary

We define *valid* leaf nodes as those meeting two requirements: a minimum of 100 clips and without extreme development/evaluation imbalance. This is a trade-off between abundant per-class data and preserving a lot of leaf nodes.[24] We take the following measures.

Merge non-valid leaf nodes with their parents. There are two variants of this process, depending on the type of branch in the hierarchy. First, non-valid siblings of valid leaf nodes are merged with their parents. In these branches, the level of specificity is fixed by the valid sibling. For instance, *Yip*, a class with few data which is sibling of *Bark* and child of *Dog*, is merged with *Dog* and the most specific label in this branch is the valid leaf *Bark*. Then, in branches without any valid leaf nodes, all leaf nodes are merged with their parents, which in turn become new leaf nodes (since they no longer have children). This process is repeated recursively, pruning the branch by moving upwards in the hierarchy, until a new leaf node becomes valid. While we ideally want to prune the branches as little as possible to preserve the most specific nodes, some low-level nodes are inevitably merged with non-specific parents, e.g., *Domestic sounds, home sounds*. The minimum data requirement is enforced at the leaf node of every branch, but not at its ancestors, which are intrinsically valid because the leaf node provides enough data. This means that, occasionally, the data explicitly associated with one ancestor may be scarce. This is due to the nomination system and annotation processes, which favour more specific labels.

[24]Given the particularities of some classes, the requirements to consider a leaf node *valid* are relaxed in a few exceptions.

Remove some valid leaf nodes to obtain a more semantically consistent vocabulary. As a result of the pruning, some parents with various children in the ontology end up having very few children in the candidate dataset. In most cases, this is not a problem as children are rather independent semantically. However, in other cases, children constitute a pre-established subset of closely related classes that makes more sense when all of them co-exist, e.g., the classes *Light engine (high frequency)*, *Medium engine (mid frequency)*, and *Heavy engine (low frequency)*, where only the former is valid. Considering the real operation of trained models, the fact that only one of these children is valid could potentially lead to unnatural predictions biased by the choice of the vocabulary. To prevent this issue, we merge some "isolated" valid leaf nodes with their parents. Hence, despite having a substantial number of light engine sounds, they are not part of the vocabulary—only *Engine* is. Note however that these more specific annotations are indeed available in the sound collection format.

Discard some intermediate nodes. This includes classes of abstract nature or with ambiguous children and few data, e.g., *Digestive* or *Arrow*, respectively. The outcome is a vocabulary of 200 classes (144 leaf nodes and 56 intermediate nodes).

B. Balancing development/evaluation sets

As a result of exhaustively labelling the evaluation set, the proportion of some frequently occurring sound events increased substantially, sometimes exceeding the number of labels in the development set. To obtain a better balance between development and evaluation sets, we first identified a set of 40 leaf nodes which benefit from transferring data from evaluation to development. Then, we selected a set of evaluation clips such that: *i)* their content encompasses mainly the 40 target classes with a minimal impact on the remaining ones—note the clips are multilabel; *ii)* they are disjoint from the remaining set of clips in terms of uploaders. Specifically, we transferred 1182 clips, resulting in an evaluation set of 10,231 clips, and a per-class development/evaluation proportion ranging from 50/50% to 75/25% in the vast majority of leaf nodes. The per-class split proportion depends on data availability, ubiquity of the sound events, degree of multilabelness of the audio clips, and non-divisibility of content from the same uploader. Exceptions

include *Chatter*, *Chirp, tweet* and *Male speech, man speaking*, for which there are more evaluation than development labels due to the exhaustive labelling of these ubiquitous events. With this transfer, we also make available some exhaustively labeled content for validation.

C. Validation Set

Some recent large audio datasets do not provide predefined validation sets [26], [129] allowing dataset users to create their own. Nonetheless, for easier dataset consumption and reproducibility we propose a candidate split of the development set into train and validation. We consider that a validation set should ideally meet the following criteria:

- **Proportion**. The validation set typically amounts to a given proportion of the development set, often between 10 and 20%. Note that due to the multilabel and variable-length nature of Freesound audio, the proportion can be different in terms of audio clips, labels, and duration.
- **Stratification**. It is usually desirable that the class label distribution is similar in both train and validation sets.
- **Contamination**. As explained in Sec. III-F, contamination across splits should be minimized.

Typical ways to make train/validation splits include random sampling or iterative stratification [72]. Both can produce desired data proportions and class distributions, the latter being popular for multilabel data.[25] However, they fail to keep non-divisibility of uploaders' content, thus generating contamination. The distribution of number of clips per uploader is very varied in the development set. However, since we already allocated a large amount of small uploaders into the evaluation set (Sec. III-F), preserving uploader non-divisibility at this point means deviating from the target class distribution. In other words, it is difficult to strictly meet the three above criteria simultaneously, hence we need to relax their application.

We focus on the contamination criteria and distinguish two types of contamination: *i) within-class* contamination (WC, when content from the same uploader *and* belonging to the same class is placed at both train and validation sets); *ii) between-class* contamination (BC, when content from the same uploader *but not* from the same class is placed at both train and validation sets). We hypothesize WC is more harmful as it could imply having the same sound source, physical location and/or recording gear in both sets. By contrast, BC would have less impact as, in most cases, the audio material would be different, and also possibly the acoustic environment. Under this hypothesis, we focus on minimizing WC contamination while being flexible with BC. To do this, we employ a method similar to that of Sec. III-F. We first define the content from one uploader labeled with the same class label as the minimum non-divisible unit. Then, we adopt an iterative process in which, after sorting the uploaders per-class appropriately, we progressively allocate their content to the validation set.

As preprocessing, we initialize the validation set with most of the data transferred from evaluation to development—this content is well suited for evaluation purposes as it is

exhaustively labeled. We then compute a score per uploader and per class. The score for uploader u in class c_i is given by:

$$\text{score}_{c_i}^u = \alpha \text{n\_labels}_{c_i}^u + \beta \frac{1}{K_u} \sum_{k=1}^{K_u} \text{n\_labels}_{c_k}^u, \quad (4)$$

where $\text{n\_labels}_{c_i}^u$ represents the number of labels provided by uploader u in class c_i, K_u is the number of classes touched by u, and α and β are tunable weights to set the relevance of each term, both $\in [0, 1]$. The first term is the amount of data in c_i by u, whereas the second term is the average number of labels per class, accounting for the scattering of u across classes. Uploaders are sorted in ascending score order and the content of low-score uploaders is transferred first. By tuning α and β we aim to promote the uploaders providing a small amount of data in the class under question, c_i, with minimal or no scattering. This facilitates the adjustment to a target class distribution while minimizing contamination (both WC and BC). This first group of uploaders is followed by others with smooth scattering across classes, avoiding uploaders with large contributions concentrated in specific classes. This again facilitates adjusting to a target distribution while minimizing the need to split content from the same uploader in one class (i.e., WC contamination), but allowing BC contamination.

Once the validation set is initialized and the uploaders are sorted per-class, we allocate data to the validation set as shown in Algorithm 2. We traverse the classes in several passes, and,

---

Algorithm 2: Data allocation to validation set

Data: Initialized validation data per-class $V = \{v_{c_i}\}_{i=1}^C$, uploaders ranking in development set per-class $U = \{\boldsymbol{u_{c_i}}\}_{i=1}^C$

1 **for** *pass* $n = 1, 2, \ldots N$ **do**
2   **for** *class* $c_i \in C$ **do**
3   get current validation target t_{c_i}
4   **while** $v_{c_i} < t_{c_i}$ **do**
5   get next uploader u in ranking $\boldsymbol{u_{c_i}}$
6          $v_{c_i} \leftarrow v_{c_i}$ + data from u in c_i
7   **if** *data is multilabel to class c_j* **then**
8             $v_{c_j} \leftarrow v_{c_j}$ + data from u in c_j
9   **end**
10   **end**
11   **end**
12 **end**

Result: A candidate validation set

---

for each class c_i, we progressively allocate content from the ranked uploaders until a target data amount t_{c_i} is reached. Note that when separating the class c_i, the algorithm does not care about a given uploader u contributing to another class c_j (BC contamination), unless there is at least one clip bearing labels for both c_i and c_j. WC contamination can be produced in lines 6 and 8. We designed the step in line 6 so that, if adding the content from u implies exceeding the validation target t_{c_i} by more than 15%, two things can happen. If the current validation amount is $v_{c_i} > 0.75 t_{c_i}$, the content is not transferred, v_{c_i} is deemed sufficient and the procedure halted for c_i. This flexibility allows the minimization of WC contamination at the expense of deteriorating stratification.

---

[25]Random sampling does not account for stratification per se, but a workaround is to compute many train/validation splits and choose the one that minimizes a distance between the respective class distributions.

Else, if $v_{c_i} <= 0.75t_{c_i}$, the content from u in c_i is split and the amount needed to reach t_{c_i} is allocated, causing WC contamination. Similar heuristics are adopted for line 8.

Using the proposed scheme, we process clips from the 4936 uploaders in the development set. Due to the high variability of users, the process needs initial debugging with a subset of classes in order to tune the weights α and β. We finally use $\alpha = 0.4$ and $\beta = 0$ when an uploader contributes only to one class, and $\alpha = 0.3$ and $\beta = 0.7$ otherwise. We use $N = 2$ passes starting from classes in need of more validation data, which allows us to reach a reasonable stratification. The target validation proportion is 15% of the development labels per-class, except for the largest 17 classes where we reduced this percentage progressively. The first-pass target is to fill 60% of the 15%-target, which is the goal in the second-pass. We only consider the leaf nodes for this process ($C = 144$). This is done for simplicity and because the leaf nodes are the most specific data that will receive labels from the rest of the ontology levels upon propagation to their ancestors. In this way, validation data at all levels of the ontology is guaranteed.

The outcome is a validation set which represents a tradeoff between stratification and contamination. Composed of 4170 audio clips, it amounts to 13.3% of the content associated with leaf nodes and 10.2% of the entire development set. Its main statistics are listed in Table VIII. Out of the 2224 uploaders with content in the validation set, 641 also have content in the train set—mostly corresponding to BC contamination. Sec. V-C describes sound event tagging experiments comparing the proposed split to others obtained via off-the-shelf approaches.

TABLE VIII
MAIN STATISTICS FOR CANDIDATE VALIDATION SET

clips	duration	uploaders
4170	9.9h	2224

D. Hierarchical Label Propagation

At this point, the labels in train, validation and evaluation sets are usually from classes corresponding to lower levels of the ontology, especially for the evaluation set (see Sec. III-G). To obtain an exhaustive labelling hierarchy-wise, we need to propagate the current labels in the upwards direction to the root of the ontology, determining the ancestors in the hierarchical path and automatically assigning them to the corresponding audio clips. This hierarchical label propagation process is sometimes referred to as *label smearing* [40], [130]. In most cases, this is straightforward as there is one single unequivocal path from a given low-level node to the root. However, in other cases, nodes and root are connected by more than one path. Among these multiple-path cases, some have all the paths valid by default according to the semantics of the node. This allows straightforward propagation as in the single-path case, e.g., *Doorbell* can be directly propagated to *Door* and *Alarm*. However, in the majority of cases, only a subset of the paths is valid (often only one path), or even none of the paths is valid by default due to the parents-node relationship. For instance, *Buzz* cannot be directly propagated to its parents *Fly, housefly* or *Bee, wasp, etc.* unless we have explicit information about

the source of the buzz sound. In these cases, we need knowledge of the correct immediate parent(s) to unambiguously infer ancestors for a complete hierarchical labelling. Parents disambiguation can be carried out in different ways depending on the annotation task. In the clips annotated only with the validation task, the disambiguating parents will exist if and only if the nomination system proposed them. For the clips annotated also with the refinement task, raters were instructed to specify the disambiguating parents when needed; however, we detected that they were not always specified.

As a result, in these cases, ancestors cannot be inferred from the leaf node, leading to hierarchical paths featuring missing parts. For example, *Growling* is connected directly to *Animal* in several cases where information of the source animal is not available. The policy followed in case of ambiguous ancestors was to not include these labels (hence potentially creating missing "Present" labels in the mid- or high-levels of the ontology) instead of possibly generating incorrect labels. In the development set, these cases are provided *as is* since it is less critical. By contrast, because the cases in the evaluation set are more critical, they were partially reviewed and corrected.

To finalize this process, we filter out labels beyond the 200 selected. In the majority of cases, these correspond to abstract or blacklisted classes of the ontology [26]. This is another reason why some clips have labels up to the ontology root while others only have a portion of the ancestors or even one single label. For example, *Whoosh, swoosh, swish* has no hierarchy as all class labels in its path were either removed previously due to specified constraints (*Arrow*) or removed in this last step (as classes above *Arrow* are abstract). This can be easily spotted in the provided ground truth CSV files.[4] The number of labels before/after the hierarchical label propagation process can be seen in Table IV (*unpropagated* and *propagated*, respectively). The outcome is a set of hierarchically propagated labels consistently encompassing all relevant levels of the ontology. Note the considerable increase of labels, despite that we are ignoring parts of the ontology. This is the final ground truth provided for FSD50K.

REFERENCES

[1] T. Drugman, J. Urbain, N. Bauwens, R. Chessini, A.-S. Aubriot, P. Lebecque, and T. Dutoit, "Audio and contact microphones for cough detection," in *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.

[2] A. Hüwel, K. Adiloğlu, and J.-H. Bach, "Hearing aid research data set for acoustic environment recognition," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 706–710.

[3] E. Messner, M. Fediuk, P. Swatek, S. Scheidl, F.-M. Smolle-Jüttner, H. Olschewski, and F. Pernkopf, "Multi-channel lung sound classification with convolutional recurrent neural networks," *Computers in Biology and Medicine*, p. 103831, 2020.

[4] J. P. Bello, C. Silva, O. Nov, R. L. DuBois, A. Arora, J. Salamon, C. Mydlarz, and H. Doraiswamy, "SONYC: A system for monitoring, analyzing, and mitigating urban noise pollution," *Communications of the ACM*, vol. 62, no. 2, pp. 68–77, 2019.

[5] J. Cramer, V. Lostanlen, A. Farnsworth, J. Salamon, and J. P. Bello, "Chirping up the right tree: Incorporating biological taxonomies into deep bioacoustic classifiers," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 901–905.

[6] V. Lostanlen, J. Salamon, A. Farnsworth, S. Kelling, and J. P. Bello, "Robust sound event detection in bioacoustic sensor networks," *PloS one*, vol. 14, no. 10, p. e0214168, 2019.

[7] K. Xu, H. Cai, X. Liu, Z. Gao, and B. Zhang, "North Atlantic Right Whale call detection with very deep convolutional neural networks," *The Journal of the Acoustical Society of America*, vol. 141, no. 5, pp. 3944–3945, 2017.

[8] Y. Wang, L. Neves, and F. Metze, "Audio-based multimedia event detection using deep recurrent neural networks," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 2742–2746.

[9] A. Jansen, J. F. Gemmeke, D. P. W. Ellis, X. Liu, W. Lawrence, and D. Freedman, "Large-scale audio event discovery in one million YouTube videos," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 786–790.

[10] M. Crocco, M. Cristani, A. Trucco, and V. Murino, "Audio surveillance: A systematic review," *ACM Computing Surveys (CSUR)*, vol. 48, no. 4, pp. 1–46, 2016.

[11] H. A. Sánchez-Hevia, D. Ayllón, R. Gil-Pita, and M. Rosa-Zurera, "Maximum likelihood decision fusion for weapon classification in wireless acoustic sensor networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1172–1182, 2017.

[12] M. Morrison and B. Pardo, "OtoMechanic: Auditory automobile diagnostics via query-by-example," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, New York University, NY, USA, October 2019, pp. 169–173.

[13] A. Mesaros, T. Heittola, E. Benetos, P. Foster, M. Lagrange, T. Virtanen, and M. D. Plumbley, "Detection and classification of acoustic scenes and events: Outcome of the DCASE 2016 challenge," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 2, pp. 379–393, 2017.

[14] P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio, and M. Vento, "Reliable detection of audio events in highly noisy environments," *Pattern Recognition Letters*, vol. 65, pp. 22–28, 2015.

[15] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, "Acoustic event detection in real life recordings," in *2010 18th European Signal Processing Conference*. IEEE, 2010, pp. 1267–1271.

[16] C. V. Cotton and D. P. W. Ellis, "Spectral vs. spectro-temporal features for acoustic event detection," in *2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2011, pp. 69–72.

[17] E. Cakir, T. Heittola, H. Huttunen, and T. Virtanen, "Polyphonic sound event detection using multi label deep neural networks," in *2015 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2015, pp. 1–7.

[18] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2015, pp. 1–6.

[19] G. Parascandolo, H. Huttunen, and T. Virtanen, "Recurrent neural networks for polyphonic sound event detection in real life recordings," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 6440–6444.

[20] E. Cakır, G. Parascandolo, T. Heittola, H. Huttunen, and T. Virtanen, "Convolutional recurrent neural networks for polyphonic sound event detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1291–1303, 2017.

[21] P. Foster, S. Sigtia, S. Krstulovic, J. Barker, and M. D. Plumbley, "CHiME-home: A dataset for sound source recognition in a domestic environment," in *2015 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2015.

[22] K. J. Piczak, "ESC: Dataset for environmental sound classification," in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2015, pp. 1015–1018.

[23] A. Mesaros, T. Heittola, and T. Virtanen, "TUT database for acoustic scene classification and sound event detection," in *24th European Signal Processing Conference 2016 (EUSIPCO 2016)*, Budapest, Hungary, 2016.

[24] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 843–852.

[25] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

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

[27] M. Cartwright, A. E. M. Mendez, J. Cramer, V. Lostanlen, G. Dove, H.-H. Wu, J. Salamon, O. Nov, and J. Bello, "SONYC urban sound tagging (SONYC-UST): A multilabel dataset from an urban acoustic sensor network," in *Proceedings of the Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE)*, October 2019, pp. 35–39.

[28] N. Turpault, R. Serizel, J. Salamon, and A. P. Shah, "Sound event detection in domestic environments with weakly labeled data and soundscape synthesis," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, New York University, NY, USA, October 2019, pp. 253–257.

[29] S. Adavanne, A. Politis, and T. Virtanen, "A multi-room reverberant dataset for sound event localization and detection," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, New York University, NY, USA, October 2019, pp. 10–14.

[30] S. Gharib, K. Drossos, E. Fagerlund, and T. Virtanen, "VOICe: A sound event detection dataset for generalizable domain adaptation," *arXiv preprint arXiv:1911.07098*, 2019.

[31] G. Dekkers, S. Lauwereins, B. Thoen, M. W. Adhana, H. Brouckxon, T. van Waterschoot, B. Vanrumste, M. Verhelst, and P. Karsmakers, "The SINS database for detection of daily activities in a home environment using an acoustic sensor network," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017)*, November 2017, pp. 32–36.

[32] E. Fonseca, M. Plakal, D. P. W. Ellis, F. Font, X. Favory, and X. Serra, "Learning sound event classifiers from web audio with noisy labels," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.

[33] E. Fonseca, M. Plakal, F. Font, D. P. W. Ellis, and X. Serra, "Audio tagging with noisy labels and minimal supervision," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, NY, USA, 2019.

[34] M. Cartwright, J. Cramer, A. E. Mendez Mendez, Y. Wang, H.-H. Wu, V. Lostanlen, M. Fuentes, G. Dove, C. Mydlarz, J. Salamon, O. Nov, and J. P. Bello, "SONYC-UST-V2: An urban sound tagging dataset with spatiotemporal context," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020)*, Tokyo, Japan, November 2020, pp. 16–20.

[35] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *European Conference on Computer Vision*. Springer, 2014, pp. 740–755.

[36] W. Li, L. Wang, W. Li, E. Agustsson, and L. Van Gool, "WebVision database: Visual learning and understanding from web data," *arXiv preprint arXiv:1708.02862*, 2017.

[37] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov *et al.*, "The Open Images dataset V4: Unified image classification, object detection,

and visual relationship detection at scale," *International Journal of Computer Vision*, vol. 128, no. 7, pp. 1956–1981, 2020.

[38] B. McFee, J. W. Kim, M. Cartwright, J. Salamon, R. M. Bittner, and J. P. Bello, "Open-source practices for music signal processing research: Recommendations for transparent, sustainable, and reproducible audio research," *IEEE Signal Processing Magazine*, vol. 36, no. 1, pp. 128–137, 2018.

[39] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2014, pp. 1041–1044.

[40] S. Hershey, D. P. W. Ellis, E. Fonseca, A. Jansen, C. Liu, R. C. Moore, and M. Plakal, "The benefit of temporally-strong labels in audio event classification," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

[41] H. Chen, W. Xie, A. Vedaldi, and A. Zisserman, "VGGSound: A large-scale audio-visual dataset," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 721–725.

[42] E. Fonseca, S. Hershey, M. Plakal, D. P. W. Ellis, A. Jansen, and R. C. Moore, "Addressing missing labels in large-scale sound event recognition using a teacher-student framework with loss masking," *IEEE Signal Processing Letters*, vol. 27, pp. 1235–1239, 2020.

[43] B. McFee, J. Salamon, and J. P. Bello, "Adaptive pooling operators for weakly labeled sound event detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 11, pp. 2180–2193, 2018.

[44] B. Barz and J. Denzler, "Do we train on test data? purging CIFAR of near-duplicates," *Journal of Imaging*, vol. 6, no. 6, p. 41, 2020.

[45] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," University of Toronto, Tech. Rep., 2009.

[46] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar, "Do ImageNet classifiers generalize to ImageNet?" in *International Conference on Machine Learning*. PMLR, 2019, pp. 5389–5400.

[47] L. Beyer, O. J. Hénaff, A. Kolesnikov, X. Zhai, and A. v. d. Oord, "Are we done with ImageNet?" *arXiv preprint arXiv:2006.07159*, 2020.

[48] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 248–255.

[49] E. Fonseca, D. Ortego, K. McGuinness, N. E. O'Connor, and X. Serra, "Unsupervised contrastive learning of sound event representations," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

[50] E. Fonseca, A. Jansen, D. P. W. Ellis, S. Wisdom, M. Tagliasacchi, J. R. Hershey, M. Plakal, S. Hershey, R. C. Moore, and X. Serra, "Self-supervised learning from automatically separated sound scenes," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2021.

[51] A. Jansen, M. Plakal, R. Pandya, D. P. W. Ellis, S. Hershey, J. Liu, R. C. Moore, and R. A. Saurous, "Unsupervised learning of semantic audio representations," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 126–130.

[52] W. Han, E. Coutinho, H. Ruan, H. Li, B. Schuller, X. Yu, and X. Zhu, "Semi-supervised active learning for sound classification in hybrid learning environments," *PloS one*, vol. 11, no. 9, p. e0162075, 2016.

[53] Y. Wang, A. E. M. Mendez, M. Cartwright, and J. P. Bello, "Active learning for efficient audio annotation and classification with a large amount of unlabeled data," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 880–884.

[54] Z. Shuyang, T. Heittola, and T. Virtanen, "An active learning method using clustering and committee-based sample selection for sound event classification," in *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE, 2018, pp. 116–120.

[55] ——, "Active learning for sound event detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2895–2905, 2020.

[56] F. Font, G. Roma, and X. Serra, "Freesound technical demo," in *Proceedings of the 21st ACM International Conference on Multimedia*. ACM, 2013, pp. 411–412.

[57] D. Stowell and M. Plumbley, "An open dataset for research on audio field recording archives: freefield1010," in *Audio Engineering Society Conference: 53rd International Conference: Semantic Audio*. Audio Engineering Society, 2014.

[58] E. Fonseca, M. Plakal, F. Font, D. P. W. Ellis, X. Favory, J. Pons, and X. Serra, "General-purpose tagging of Freesound audio with AudioSet labels: task description, dataset, and baseline," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, 2018.

[59] E. Fonseca, J. Pons, X. Favory, F. Font, D. Bogdanov, A. Ferraro, S. Oramas, A. Porter, and X. Serra, "Freesound Datasets: a platform for the creation of open audio datasets," in *Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR 2017)*, Suzhou, China, 2017, pp. 486–493.

[60] M. F. Porter *et al.*, "An algorithm for suffix stripping." *Program*, vol. 14, no. 3, pp. 130–137, 1980.

[61] R. Serizel, N. Turpault, A. Shah, and J. Salamon, "Sound event detection in synthetic domestic environments," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 86–90.

[62] D. Hendrycks, K. Zhao, S. Basart, J. Steinhardt, and D. Song, "Natural adversarial examples," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 262–15 271.

[63] S. Wisdom, H. Erdogan, D. P. Ellis, R. Serizel, N. Turpault, E. Fonseca, J. Salamon, P. Seetharaman, and J. R. Hershey, "What's all the FUSS about free universal sound separation data?" in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 186–190.

[64] M. Cartwright, A. Seals, J. Salamon, A. Williams, S. Mikloska, D. MacConnell, E. Law, J. P. Bello, and O. Nov, "Seeing sound: Investigating the effects of visualizations and complexity on crowdsourced audio annotations," *Proceedings of the ACM on Human-Computer Interaction*, vol. 1, no. CSCW, pp. 1–21, 2017.

[65] EBU Recommendation R 128, "Loudness Normalisation and Permitted Maximum Level of Audio Signals," *European Broadcasting Union*, 2014.

[66] M. Sabou, K. Bontcheva, L. Derczynski, and A. Scharl, "Corpus annotation through crowdsourcing: Towards best practice guidelines." in *LREC*, 2014, pp. 859–866.

[67] M. Meire, P. Karsmakers, and L. Vuegen, "The impact of missing labels and overlapping sound events on multi-label multi-instance learning for sound event classification," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, NY, USA, 2019.

[68] T. Virtanen, M. D. Plumbley, and D. P. W. Ellis, *Computational Analysis of Sound Scenes and Events*. Springer, 2018.

[69] B. Whitman, G. Flake, and S. Lawrence, "Artist detection in music with minnowmatch," in *Neural Networks for Signal Processing XI: Proceedings of the 2001 IEEE Signal Processing Society Workshop (IEEE Cat. No. 01TH8584)*. IEEE, 2001, pp. 559–568.

[70] M. I. Mandel and D. P. W. Ellis, "Song-level features and support vector machines for music classification," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR 2005)*, 2005.

[71] A. Flexer and D. Schnitzer, "Album and artist effects for audio similarity at the scale of the web," *Children*, vol. 15, no. 15.95, pp. 4–07, 2009.

[72] K. Sechidis, G. Tsoumakas, and I. Vlahavas, "On the stratification of multi-label data," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2011, pp. 145–158.

[73] P. Toth and S. Martello, *Knapsack Problems: Algorithms and Computer Implementations*. Wiley, 1990.

[74] X. Favory, E. Fonseca, F. Font, and X. Serra, "Facilitating the manual annotation of sounds when using large taxonomies," in *Proceedings of the 23rd Conference of Open Innovations Association, FRUCT*, no. 23. FRUCT Oy, 2018, pp. 447–451.

[75] K.-H. Cheng, S.-Y. Chou, and Y.-H. Yang, "Multi-label few-shot learning for sound event recognition," in *2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2019, pp. 1–5.

[76] J. Salamon, D. MacConnell, M. Cartwright, P. Li, and J. P. Bello, "Scaper: A library for soundscape synthesis and augmentation," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2017, pp. 344–348.

[77] A. Shah, A. Kumar, A. G. Hauptmann, and B. Raj, "A closer look at weak label learning for audio events," *CoRR*, vol. abs/1804.09288, 2018.

[78] N. Turpault, R. Serizel, and E. Vincent, "Limitations of weak labels for embedding and tagging," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 131–135.

[79] V. Morfi and D. Stowell, "Data-efficient weakly supervised learning for low-resource audio event detection using deep learning," in *Pro-*

ceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018).

[80] G. Lafay, E. Benetos, and M. Lagrange, "Sound event detection in synthetic audio: Analysis of the DCASE 2016 task results," in 2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), Oct 2017, pp. 11–15.

[81] A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj, and T. Virtanen, "DCASE 2017 Challenge Setup: Tasks, datasets and baseline system," in Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017), November 2017, pp. 85–92.

[82] L. Malfait, J. Berger, and M. Kastner, "P.563—The ITU-T standard for single-ended speech quality assessment," IEEE Transactions on Audio, Speech, and Language Processing, vol. 14, no. 6, pp. 1924–1934, 2006.

[83] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," IEEE Signal Processing Letters, vol. 24, no. 3, pp. 279–283, 2017.

[84] J. Li, M. L. Seltzer, X. Wang, R. Zhao, and Y. Gong, "Large-scale domain adaptation via teacher-student learning," arXiv preprint arXiv:1708.05466, 2017.

[85] C. Northcutt, L. Jiang, and I. Chuang, "Confident learning: Estimating uncertainty in dataset labels," Journal of Artificial Intelligence Research, vol. 70, pp. 1373–1411, 2021.

[86] H. Lei, J. Choi, A. Janin, and G. Friedland, "User verification: Matching the uploaders of videos across accounts," in 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).   IEEE, 2011, pp. 2404–2407.

[87] E. Fonseca, A. Ferraro, and X. Serra, "Improving sound event classification by increasing shift invariance in convolutional neural networks," arXiv preprint arXiv:2107.00623, 2021.

[88] E. Cakir, E. C. Ozan, and T. Virtanen, "Filterbank learning for deep neural network based polyphonic sound event detection," in 2016 International Joint Conference on Neural Networks (IJCNN).   IEEE, 2016, pp. 3399–3406.

[89] H. Park and C. D. Yoo, "CNN-based learnable gammatone filterbank and equal-loudness normalization for environmental sound classification," IEEE Signal Processing Letters, vol. 27, pp. 411–415, 2020.

[90] E. Fonseca, F. Font, and X. Serra, "Model-agnostic approaches to handling noisy labels when training sound event classifiers," in 2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA).   IEEE, 2019, pp. 16–20.

[91] B. Elizalde, S. Zarar, and B. Raj, "Cross modal audio search and retrieval with joint embeddings based on text and audio," in 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).   IEEE, 2019, pp. 4095–4099.

[92] X. Favory, K. Drossos, T. Virtanen, and X. Serra, "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, 2020.

[93] H. Shrivaslava, Y. Yin, R. R. Shah, and R. Zimmermann, "Mt-Gcn for multi-label audio tagging with noisy labels," in 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).   IEEE, 2020, pp. 136–140.

[94] A. Jati, N. Kumar, R. Chen, and P. Georgiou, "Hierarchy-aware loss function on a tree structured label space for audio event detection," in 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).   IEEE, 2019, pp. 6–10.

[95] A. Kumar and B. Raj, "Audio event and scene recognition: A unified approach using strongly and weakly labeled data," in 2017 International Joint Conference on Neural Networks (IJCNN).   IEEE, 2017, pp. 3475–3482.

[96] S. Gharib, K. Drossos, E. Cakir, D. Serdyuk, and T. Virtanen, "Unsupervised adversarial domain adaptation for acoustic scene classification," in Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018), November 2018, pp. 138–142.

[97] D. Bogdanov, A. Porter, H. Boyer, X. Serra et al., "Cross-collection evaluation for music classification tasks," in Proceedings of the 17th International Society for Music Information Retrieval Conference (IS-MIR).   ISMIR, 2016.

[98] X. Favory, F. Font, and X. Serra, "Search result clustering in collaborative sound collections," in Proceedings of the 2020 International Conference on Multimedia Retrieval, 2020, pp. 207–214.

[99] I. Kavalerov, S. Wisdom, H. Erdogan, B. Patton, K. Wilson, J. Le Roux, and J. R. Hershey, "Universal sound separation," in 2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), October 2019, pp. 175–179.

[100] J. Abeßer, "USM-SED: A dataset for polyphonic sound event detection in urban sound monitoring scenarios," arXiv preprint arXiv:2105.02592, 2021.

[101] N. Turpault, S. Wisdom, H. Erdogan, J. R. Hershey, R. Serizel, E. Fonseca, P. Seetharaman, and J. Salamon, "Improving sound event detection in domestic environments using sound separation," in Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020), Tokyo, Japan, November 2020, pp. 205–209.

[102] D. A. Van Leeuwen and N. Brümmer, "An introduction to application-independent evaluation of speaker recognition systems," in Speaker classification I.   Springer, 2007, pp. 330–353.

[103] Ç. Bilen, G. Ferroni, F. Tuveri, J. Azcarreta, and S. Krstulović, "A framework for the robust evaluation of sound event detection," in 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).   IEEE, 2020, pp. 61–65.

[104] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold et al., "CNN architectures for large-scale audio classification," in 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).   IEEE, 2017, pp. 131–135.

[105] Scikit-learn documentation 0.23.2. Precision-recall. [Online]. Available: https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html

[106] ——. Metrics and scoring: quantifying the quality of predictions. [Online]. Available: https://scikit-learn.org/stable/modules/model_evaluation.html#precision-recall-f-measure-metrics

[107] VLFeat.org. Plotting AP and ROC curves. [Online]. Available: https://www.vlfeat.org/overview/plots-rank.html

[108] D. M. Green, J. A. Swets et al., Signal Detection Theory and Psychophysics.   Wiley New York, 1966, vol. 1.

[109] Martín Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[110] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in 3rd International Conference on Learning Representations, ICLR, 2015.

[111] J. Davis and M. Goadrich, "The relationship between Precision-Recall and ROC curves," in Proceedings of the 23rd International Conference on Machine Learning, 2006, pp. 233–240.

[112] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "PANNs: Large-scale pretrained audio neural networks for audio pattern recognition," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 2880–2894, 2020.

[113] E. Fonseca, R. Gong, D. Bogdanov, O. Slizovskaia, E. Gómez, and X. Serra, "Acoustic scene classification by ensembling gradient boosting machine and convolutional neural networks," in Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017), November 2017, pp. 37–41.

[114] E. Fonseca, R. Gong, and X. Serra, "A simple fusion of deep and shallow learning for acoustic scene classification," in Proceedings of the 15th Sound and Music Computing Conference, Limassol, Cyprus, 2018.

[115] A. Pérez-López, E. Fonseca, and X. Serra, "A hybrid parametric-deep learning approach for sound event localization and detection," in Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019), New York University, NY, USA, October 2019, pp. 189–193.

[116] J. Ebbers and R. Häb-Umbach, "Convolutional recurrent neural network and data augmentation for audio tagging with noisy labels and minimal supervision," in Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019), New York University, NY, USA, October 2019, pp. 64–68.

[117] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in Proceedings of the 32nd International Conference on Machine Learning, 2015, pp. 448–456.

[118] M. Dorfer and G. Widmer, "Training general-purpose audio tagging networks with noisy labels and iterative self-verification," in Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018), 2018, pp. 178–182.

[119] B. Kim and B. Pardo, "Sound event detection using point-labeled data," in 2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA).   IEEE, 2019.

[120] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in 3rd International Conference on Learning Representations, ICLR, 2015.

[121] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[122] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700–4708.

[123] T. Iqbal, Y. Cao, Q. Kong, M. D. Plumbley, and W. Wang, "Learning with out-of-distribution data for audio classification," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 636–640.

[124] M. Won, A. Ferraro, D. Bogdanov, and X. Serra, "Evaluation of CNN-based automatic music tagging models," in *Proceedings of the 17th Sound and Music Computing Conference*, 2020.

[125] L. Ford, H. Tang, F. Grondin, and J. Glass, "A deep residual network for large-scale acoustic scene analysis," *Proceedings of Interspeech 2019*, pp. 2568–2572, 2019.

[126] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 4580–4584.

[127] H. Liu and H. Motoda, "On issues of instance selection," *Data Mining and Knowledge Discovery*, vol. 6, no. 2, p. 115, 2002.

[128] S. G. Jantzen, B. J. Sutherland, D. R. Minkley, and B. F. Koop, "Go trimming: Systematically reducing redundancy in large gene ontology datasets," *BMC research notes*, vol. 4, no. 1, p. 267, 2011.

[129] A. Mesaros, T. Heittola, and T. Virtanen, "A multi-device dataset for urban acoustic scene classification," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, November 2018, pp. 9–13.

[130] J. Gao, Z. Li, R. Nevatia *et al.*, "Knowledge concentration: Learning 100k object classifiers in a single CNN," *arXiv preprint arXiv:1711.07607*, 2017.
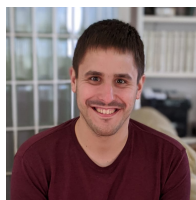
Jordi Pons is a researcher at Dolby Laboratories. Previously, he received a PhD in music technology, large-scale audio collections, and deep learning at the Music Technology Group (Universitat Pompeu Fabra, Barcelona). He also recieved a MSc in sound and music computing (Universitat Pompeu Fabra, Barcelona), and his BSc was in telecommunications engineering (Universitat Politècnica de Catalunya, Barcelona). He also interned at IRCAM (Paris), at the German Hearing Center (Hannover), at Pandora Radio (USA, Bay Area), and at Telefónica Research (Barcelona).
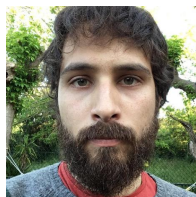
Frederic Font is a senior researcher at the Music Technology Group of the Department of Information and Communication Technologies of Universitat Pompeu Fabra, Barcelona, Spain. He received the MSc and PhD degrees in Sound and Music Computing in 2010 and 2015, respectively, from Universitat Pompeu Fabra. His current research is focused on the understanding and analysis of large audio collections, including sound characterisation and classification, to improve sound retrieval techniques and to facilitate the reuse of large audio collections in creative and scientific contexts. Frederic is the coordinator of the Freesound website and related research and development projects, and has recently coordinated the EU funded Audio Commons Initiative.

Eduardo Fonseca is a researcher with the Music Technology Group, Universitat Pompeu Fabra, where he received his Ph.D. degree under the supervision of Dr. Xavier Serra, partially funded by two Google Faculty Research Awards. His research interests include audio dataset creation and learning algorithms for sound event recognition using different types of supervision, including learning with noisy labels and self-supervision. He received two M.Sc. degrees in Acoustics Engineering from Aalborg University and in Telecommunications Engineering from the Technical University of Madrid. He has interned at Google Research in 2019 and 2020 and his work received one of the Best Paper Awards at WASPAA21. He is actively involved in the DCASE community, where he has served as a Challenge Task Organizer and Technical Program Co-Chair.

Xavier Serra is a Professor of the Department of Information and Communication Technologies and Director of the Music Technology Group at the Universitat Pompeu Fabra in Barcelona. After a multidisciplinary academic education, he obtained a PhD in Computer Music from Stanford University in 1989 with a dissertation on the spectral processing of musical sounds that is considered a key reference in the field. His research interests cover the computational analysis, description, and synthesis of sound and music signals, with a balance between basic and applied research and approaches from both scientific/technological and humanistic/artistic disciplines. Dr. Serra is very active in the fields of Audio Signal Processing, Sound and Music Computing, Music Information Retrieval and Computational Musicology at the local and international levels, being involved in the editorial board of a number of journals and conferences and giving lectures on current and future challenges of these fields. He was awarded an Advanced Grant from the European Research Council to carry out the project CompMusic aimed at promoting multicultural approaches in music information research.

Xavier Favory graduated from Ecole Nationale Supérieure de l'Electronique et de ses Applications (France) with an Electronic Engineering Master Degree, specializing in Multimedia Systems in 2014. He then obtained the Master's degree in Acoustics, Signal Processing and Informatics applied to Music from IRCAM, University Pierre and Marie Curie and Télécom ParisTech in 2015. He earned his Ph.D. in 2021, at the Music Technology Group, Unversitat Pompeu Fabra, where he is now continuing as a post-doctoral researcher. His research interests lie in the areas of audio signal processing, machine learning and human-computer interaction. He is author and co-author of several peer-reviewed publications, and he has been contributing as a reviewer for multiple conferences or workshops (e.g., International Society for Music Information Retrieval, Web Audio Conference, Digital Audio Effects Conference, International Joint Conference on Neural Networks). He also has strong interests in web technologies and development.