# SOUND EFFECTS DATASET UNIFICATION WITH THE UNIVERSAL CATEGORY SYSTEM

*Jun Woo Beck and Alexander Lerch*

Music Informatics Group
Georgia Institute of Technology
Atlanta, GA, USA
jbeck85@gatech.edu alexander.lerch@gatech.edu

## ABSTRACT

Sound effects (SFX) datasets and libraries often employ distinct tagging schemes, taxonomies, and metadata structures. This creates challenges for research on SFX classification and generation because incompatible taxonomies lead to siloed datasets that might require individualized approaches, result in non-comparable outcomes, and prevent data merging strategies. We propose a modular dataset relabeling framework that adopts the Universal Category System (UCS), an industry-standard hierarchical taxonomy for sound effects, as a shared structural foundation. This open-source framework enables us (i) to convert tags of existing datasets to UCS with a rule-based multi-stage pipeline and conflict resolution to achieve high automatic conversion rates, (ii) to suggest a stratified dataset split for the new labels, and (iii) to combine multiple datasets. To showcase the practical utility, we introduce the *EnvSound-UCS* dataset, a publicly available unified UCS-compliant dataset of environmental sounds with 58,057 sound clips from three sources: AudioSet, FSD50K, and ESC-50.

## 1. INTRODUCTION

The rapid growth and ever-increasing complexity of modern audio-focused artificial intelligence and machine learning approaches results in an increased need for large-scale audio datasets. While there is a variety of datasets available for machine learning tasks in the audio domain, each dataset tends to employ its own tagging vocabulary, metadata schema, and data partitioning strategy. This creates three practical challenges when working with multiple datasets:

(i) *compatibility & interoperability*: as the datasets use different, possibly inconsistent taxonomies and vocabularies, neither a meaningful comparison nor the combination of datasets into a larger dataset is possible, and different folder structures, file formats, and ways to store annotations complicate handling a large number of datasets,

(ii) *extensibility*: inconsistent, and in some cases ill-documented annotation methodologies make the extension of existing datasets tedious and complicated to achieve,

(iii) *coherence*: without a shared category space, extracting semantically consistent subsets from large datasets or assessing coverage gaps across datasets is not straightforward.

There exist efforts to address these dataset challenges for a variety of tasks from the data side, from the data loader side, or from the model side. Liu et al. combined various datasets for piano transcription into the larger dataset ACPAS [1]. Bittner et al. proposed a solution at the data loading stage with a software capable of loading annotations and data from a variety of datasets into a consistent format [2]. Ma and Lerch attempted to address the diversity of annotations through training a generalized representation for the classification of effects that can be relatively easily adapted to a multitude of annotation schemata [3]. The challenges of transparency and documentation have inspired proposals to utilize datasheets, a standardized documentation framework covering motivation, composition, acquisition, pre-processing, intended uses, distribution, and maintenance [4].

Sound effects (SFX)—audio content other than speech and music that is used in media production to support storytelling, convey environments, and enhance narrative immersion [5, 6]—encompass an exceptionally broad acoustic domain, from environmental ambiences and Foley recordings to designed synthetic textures. Unlike speech or music, whose classification systems have been standardized for over a century (e.g., the International Phonetic Alphabet [7]), no comparable consensus existed for SFX until the recent adoption of the Universal Category System [8]. Professional sound libraries have consequently relied on proprietary naming schemes, while academic datasets have independently developed their own ontologies. Cano et al. [9] proposed an MPEG-7 and WordNet-based scheme for SFX library management, observing that production-scale taxonomies require thousands of categories that do not follow a clean hierarchical structure. This fragmentation between production workflows and research communities makes SFX particularly noteworthy in the context of the dataset challenges outlined above.

The incompatibility of existing SFX dataset annotations is evident across prominent datasets. Datasets such as FSD50K [10] (51,197 clips, 200 labels), AudioSet [11] (527 classes), ESC-50 [12] (50 classes, 2,000 clips), UrbanSound8K [13] (10 classes, 8,732 clips), and Clotho [14] (caption-based) each employ distinct labeling conventions: a dog barking sound is labeled `dog_bark` in ESC-50, `Bark` in FSD50K, and `Dog` in AudioSet—three different representations of the same acoustic event. Stamatiadis et al. proposed SALT [15], a standardized label taxonomy for unifying audio event datasets with a flat label set. Anastasopoulou et al. introduced the Broad Sound Taxonomy (BST) [16], a general-purpose two-level hierarchy with 5 top-level classes—including music, speech, and sound effects—and 23 second-level classes. As BST treats sound effects as a single top-level class, it lacks the granularity needed for SFX-specific annotation.

It is noteworthy, however, that the diversity and inconsistency of annotations in commercial libraries has driven the industry, independently of academic efforts, to converge on the Universal Category System (UCS) [8] as the dominant unified standard in sound design

and post-production. UCS defines a hierarchical structure comprising 82 categories, 453 subcategories, and 9,972 synonyms, and has been widely adopted by major commercial sound libraries, enabling direct compatibility between research and production workflows.

Despite this widespread adoption in professional workflows, UCS has not found much traction in academic settings.

We propose to leverage this de-facto annotation standard and utilize publicly available data to introduce a pipeline to convert existing, proprietarily labeled SFX datasets to UCS compliant, compatible, and interoperable datasets. By establishing a shared label space, the framework enables researchers to combine multiple SFX sources into larger training corpora, extract focused subsets for targeted experiments, and seamlessly incorporate new datasets—effectively treating SFX datasets as interoperable modules rather than fixed, monolithic units. The main contributions of this work are:

(i) a rule-based tag-to-UCS conversion framework with high automatic conversion rates minimizing the need for human interaction,

(ii) a UCS-aware data merge and split tool for ensuring stratified splits and multi-source merging,

(iii) a new unified UCS compliant dataset *EnvSound-UCS* integrating three existing datasets.

The source code, the converted datasets, and the new combined dataset are publicly available.[1]

## 2. PROPOSED FRAMEWORK

The goal of this work is to design a reliable conversion pipeline for existing SFX datasets with reduced human intervention by utilizing the synonyms defined by UCS. The main objectives are (i) to seamlessly integrate multiple existing datasets with incompatible annotations by unifying the label taxonomy, (ii) to allow for easy extensibility of the new dataset, and (iii) to take advantage of the hierarchical taxonomy to allow for easy subsetting of the combined data.

For instance, this framework enables compiling a dataset for an animal sound classifier by combining ESC-50's animal classes, FSD50K's `ANIMALS` category, and AudioSet's animal-related labels into a single unified corpus—a task that previously required manual tag reconciliation.

### 2.1. CSV-Based Tag-to-UCS Conversion Pipeline

The overall process for converting tag annotations to UCS (sub)-categories is illustrated in Figure 1. Each input file carries one or more tags. The pipeline classifies each tag independently through a four-stage cascade, collects the results, and then resolves conflicts at the file level when different tags map to different categories.

#### *2.1.1. Per-tag classification*

Each input tag is first normalized (lowercased, underscores replaced by spaces) and then passed through a four-stage cascade. The stages

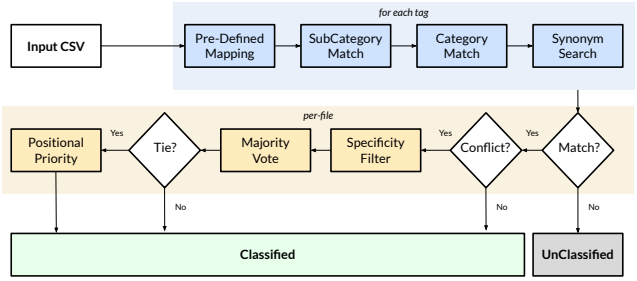


**Figure 1:** Tag-to-UCS conversion pipeline. *Blue region (for each tag):* each tag passes through a four-stage cascade; the first matching stage classifies the tag. *Yellow region (per file):* if at least one tag matched, the file proceeds to conflict resolution; otherwise it is marked unclassified.

are attempted in order; as soon as one stage produces a match, the tag is considered classified and the remaining stages are skipped:

1. **Pre-defined Mapping**: a curated mapping table converts tags with dataset-specific conventions to UCS labels (e.g., `gunshot_and_gunfire` → `GUNS/GUNSHOT`).
2. **SubCategory match**: the tag is matched against UCS subcategory names; if matched, the parent category is automatically derived from the UCS hierarchy.
3. **Category match**: the tag is matched directly against UCS category names, yielding a category-level assignment without a subcategory.
4. **Synonym match**: the tag is matched against the UCS synonym table (9,972 entries) via reverse lookup; if matched, both the category and subcategory are derived.

If none of the four stages produces a match, the tag remains unclassified.

A file is classified as long as *at least one* of its tags matches; it is marked unclassified only when *none* of its tags match at any stage. Tags that failed to match are retained in the metadata but do not influence the category assignment.

#### *2.1.2. Per-file conflict resolution*

If all matched tags for a file agree on the same UCS category, that category is assigned directly. If matched tags disagree—i.e., different tags mapped to different categories—three rules are applied in sequence to select the final assignment:

1. **Specificity filter**: tags that matched at the subcategory level are prioritized over category-only matches, as a subcategory match carries more precise semantic information. For example, if one tag maps to `GUNS/GUNSHOT` (subcategory level) and another maps to `DESIGNED` (category level only), the subcategory-level match is preferred.
2. **Majority vote**: among the remaining matched tags, the category supported by the largest number of tags is selected.
3. **Positional priority**: if a tie remains, the category associated with the last (rightmost) tag in the original annotation list is chosen. In FSD50K, where label propagation places

[1] Code and data:
`https://github.com/JunWooBeck/ucs-sfx-tools`
`https://github.com/JunWooBeck/fsd50k-ucs`
`https://github.com/JunWooBeck/audioset-ucs`
`https://github.com/JunWooBeck/esc50-ucs`
`https://github.com/JunWooBeck/envsound-ucs`

broader tags after specific descriptors, later tags tend to represent the primary acoustic content rather than incidental sounds. For example, a clip tagged `Coin_dropping, Singing, Human_voice` is assigned to VOICES rather than FOLEY. AudioSet uses a different tag ordering convention; the applicability of this heuristic to AudioSet is acknowledged as a limitation (Section 6.1). All files where this rule was applied are added to the ambiguity review list.

Every file with at least one matched tag receives a final category assignment through these rules; no file is left without a category. Files where matched tags produced competing categories are additionally logged to an ambiguity review list, which serves as an optional reference for manual verification.

#### *2.1.3. Auxiliary outputs*

The pipeline produces three auxiliary outputs: a list of unclassified files (i.e., files where all tags failed to match at any cascade stage), a frequency-ranked summary of unclassified tags, and the ambiguity review list. High-frequency unclassified tags might be added by users as entries to the pre-defined mapping table in the configuration file to improve the conversion rate in a re-run.

### 2.2. UCS-aware dataset splitting

Existing, previously proposed dataset splits do not account for the distribution within UCS categories. Using such existing splits might result in considerably different class distributions and imbalances between train and test sets. In the worst case, a training or test set class might have no entries. Therefore, defining new splits for the UCS-annotated data is necessary. We provide a re-splitting tool that uses a composite stratification key formed by concatenating the Category and SubCategory fields (`Category||SubCategory`). This preserves the joint distribution across both hierarchical levels. The tool employs a two-pass splitting strategy: the first pass separates train+validation from test using stratified sampling, and the second pass divides train+validation into train and validation sets. Category-SubCategory combinations with fewer than five samples cannot support stratified splitting and are therefore assigned entirely to the training set to prevent class absence in test or validation sets. The splits are validated by checking for test/validation-only classes and computing the Pearson correlation between training and test category distributions, requiring $r > 0.99$ to confirm that the split preserves the original class balance. Crucially, the tool supports merging multiple CSV files before splitting, directly enabling the aggregation of multiple datasets. Default ratios are 70/15/15 (train/validation/test) with a fixed random seed for reproducibility. The splitting tool additionally supports category-level filtering and multi-source merging through a JSON configuration file, enabling both aggregation of multiple datasets and extraction of targeted subsets (e.g., specifying only `WEATHER`, `WIND`, and `RAIN` to produce a focused outdoor ambience corpus). Each file retains a `source_dataset` field identifying its origin and preserves all original tags in the metadata, enabling post-hoc analysis of per-source contributions.

## 3. CONVERSION RESULTS

Table 1 summarizes the tag-to-UCS mapping results for the three well-known datasets FSD50K, AudioSet, and ESC-50. We process FSD50K (dev and eval splits), AudioSet (balanced training and evaluation segments), and ESC-50. The pipeline maps 100% of FSD50K and ESC-50 files and 98.49% of AudioSet files to UCS categories. Because each file may carry multiple tags, a single successful tag match is sufficient to map the entire file; this multi-tag design accounts for the high file-level mapping rates despite individual tags sometimes failing to match. ESC-50, with its compact 50-class vocabulary, achieves complete mapping without ambiguity.

**Table 1:** Tag-to-UCS conversion results. AudioSet refers to the balanced training and evaluation segments only.

| Dataset | Total | Classified | Rate | Ambiguous |
|---|---|---|---|---|
| FSD50K | 51,197 | 51,197 | 100% | 8,086 (15.8%) |
| AudioSet | 33,268 | 32,767 | 98.49% | 9,160 (28.0%) |
| ESC-50 | 2,000 | 2,000 | 100% | 0 |

Ambiguous files—those where multiple tags mapped to competing categories and required conflict resolution—account for 15.8% of FSD50K and 28.0% of AudioSet. All ambiguous files receive a final category assignment through the conflict resolution rules described in Section 2.1.2; the ambiguity review list is provided for optional manual inspection. AudioSet's higher ambiguity rate reflects the fact that nearly every clip carries broad secondary tags such as *Speech* and *Music* regardless of the primary acoustic content, producing formal category conflicts that are resolved by the three-rule conflict resolution.

The 501 unmapped AudioSet files (1.51%) contain only tags that either describe recording context rather than acoustic content (e.g., `Environmental_noise`, `Heavy_engine`) or use compound labels not covered by the mapping table (e.g., `Chopping_(food)`, `Change_ringing_(bellringing)`). These tags could be considered for addition to the mapping table for future conversion runs.

At the file level, the mapping table (Stage 1) resolves the vast majority of files: 97.0% for FSD50K and 91.0% for AudioSet. Adding subcategory matching (Stage 2) raises coverage to 98.5% and 96.1%, respectively. Category matching (Stage 3) and synonym matching (Stage 4) capture the remaining files, bringing FSD50K to 100% and AudioSet to 98.5% file-level coverage. ESC-50 achieves 100% mapping, with 86% of files resolved at Stage 1 and the remainder via synonym matching (Stage 4). These results confirm that the curated mapping table is the single most impactful component of the pipeline, while the automated matching stages provide incremental but essential coverage for long-tail tags.

Figure 2 compares the UCS category distributions of the three datasets. The long-tailed distribution is evident: `MUSICAL` alone accounts for over 20,000 files across FSD50K and AudioSet, while the smallest categories contain fewer than 700 files. FSD50K and AudioSet show complementary coverage patterns, and ESC-50's 26 categories are a strict subset of both larger datasets.

### 3.1. EnvSound-UCS: Environmental Sound Corpus

To demonstrate the framework's practical capabilities and to provide consistent multi-source data for future research into SFX classification and generation, we present an environmental sound corpus by combining ESC-50, FSD50K, and AudioSet: *EnvSound-UCS*.

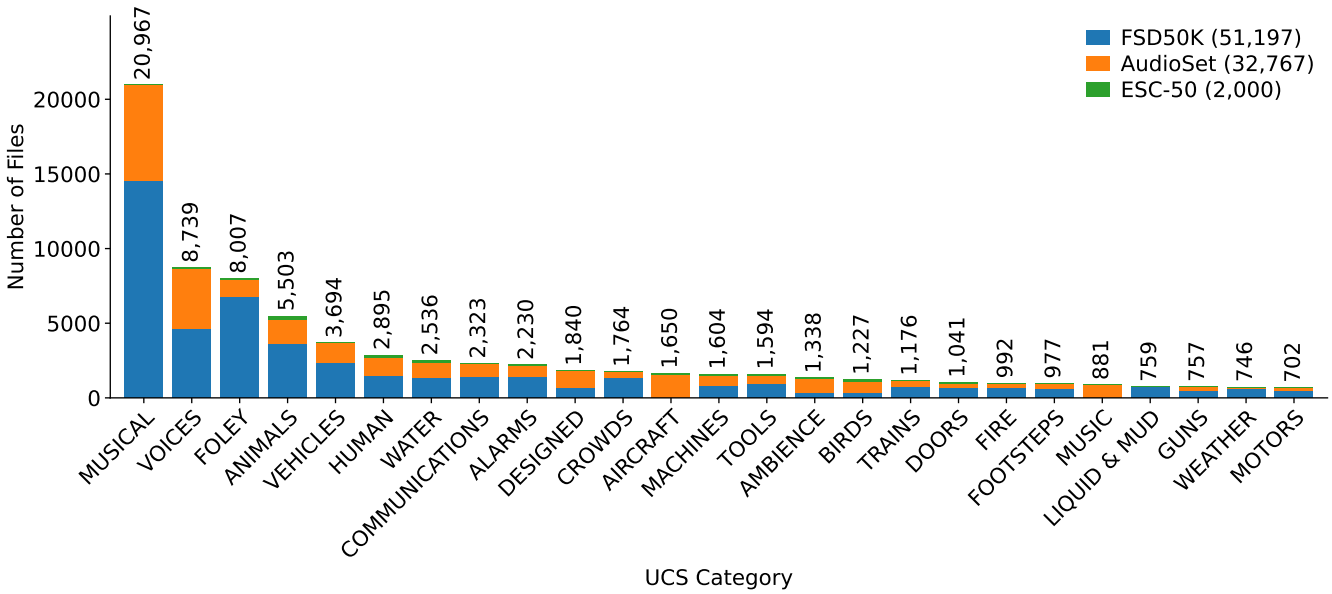


**Figure 2:** UCS category distribution across FSD50K, AudioSet, and ESC-50 (top 25 categories by combined count). Each bar shows the per-source contribution.

#### *3.1.1. Configuration*

The same environmental sound filter is applied independently to each source dataset: the `MUSICAL` category is excluded (instrumental sounds not typically considered environmental), and the `VOICES` category is restricted to the `CRYING` and `LAUGH` subcategories only, as these correspond to non-verbal vocal events present in ESC-50 (`crying_baby`, `laughing`). The `MUSIC` category (referring to recorded background music, distinct from `MUSICAL` which covers instrument performance) is retained. ESC-50, being an environmental sound dataset, contains no `MUSICAL` files and is therefore unaffected by the filter.

After filtering, each source is independently split into 70/15/15 train/validation/test partitions using the stratified splitting tool (Section 2.2). The per-source splits are then concatenated to form the final EnvSound-UCS corpus. This split-then-merge strategy ensures that each source's internal distribution is preserved and, crucially, that the individual filtered datasets (FSD50K-env, AudioSet-env) can serve as standalone benchmarks for direct comparison with the combined corpus.

#### *3.1.2. Corpus statistics*

From a combined pool of 85,964 files, filtering produces 58,057 samples spanning 59 UCS categories: 33,065 from FSD50K (after `MUSICAL` removal), 22,992 from AudioSet (after `MUSICAL` removal), and 2,000 from ESC-50.

Table 2 breaks down the per-category contributions from each source. FSD50K and AudioSet exhibit complementary coverage: for example, `ANIMALS` has 3,641 files in FSD50K but only 1,622 in AudioSet, while `DESIGNED` contains 669 FSD50K files but 1,171 from AudioSet. Of the 59 categories, 33 have no ESC-50 representation (e.g., `COMMUNICATIONS`, `DESIGNED`, `CROWDS`), meaning that aggregation with larger datasets is the only way to obtain coverage in these domains.

**Table 2:** Source contributions for EnvSound-UCS (top 10 categories by sample count).

| Category | ESC | FSD | AS | Total |
|---|---|---|---|---|
| `FOLEY` | 120 | 6,814 | 1,073 | 8,007 |
| `ANIMALS` | 240 | 3,641 | 1,622 | 5,503 |
| `VEHICLES` | – | 2,367 | 1,327 | 3,694 |
| `HUMAN` | 200 | 1,444 | 1,251 | 2,895 |
| `WATER` | 160 | 1,336 | 1,040 | 2,536 |
| `COMMUNICATIONS` | – | 1,419 | 904 | 2,323 |
| `ALARMS` | 80 | 1,422 | 728 | 2,230 |
| `DESIGNED` | – | 669 | 1,171 | 1,840 |
| `VOICES` | 80 | 1,065 | 654 | 1,799 |
| `CROWDS` | – | 1,349 | 415 | 1,764 |
| *All 59 cats* | 2,000 | 33,065 | 22,992 | 58,057 |

## 4. BENCHMARK EXPERIMENT

To establish the new dataset(s), we present benchmark results for the new categories and the new splits, as well as for the new combined environmental sounds dataset. We provide a detailed analysis of the results within the hierarchical label structure.

### 4.1. Experiments

We structure the benchmark into three experiments. The number of categories (C) and subcategories (S) varies across datasets (reported in the table headers). All models are trained and evaluated on 70/15/15 stratified splits. The framework's unified label space and provenance tracking enable controlled cross-source comparisons.

**Exp. 1: Flat classification.** We train a direct category classifier ($Cat$) and a flat subcategory classifier ($SubCat_{flat}$). From the subcategory predictions, we derive category-level results ($Cat_{flat}$)

**Table 3:** Macro F1, mean ± std (5 seeds) on original datasets for their dataset-specific 70/15/15 split. Header notation (C/S) indicates the number of categories C and subcategories S.

| | FSD50K (37/59) | AudioSet (60/158) | ESC-50 (26/20) |
|---|---|---|---|
| $Cat$ | $.52_{\pm.00}$ | $.42_{\pm.00}$ | $.89_{\pm.00}$ |
| $Cat_{flat}$ | $.71_{\pm.00}$ | $.56_{\pm.00}$ | $.99_{\pm.00}$ |
| $SubCat_{flat}$ | $.65_{\pm.00}$ | $.49_{\pm.00}$ | $.95_{\pm.01}$ |
| $SubCat_{hier}$ | $.60_{\pm.00}$ | $.42_{\pm.00}$ | $.95_{\pm.01}$ |
| $SubCat_{hier,orac}$ | $.86_{\pm.00}$ | $.74_{\pm.01}$ | $.95_{\pm.01}$ |

by mapping predicted subcategories to their parent categories. Each dataset uses its own train/test split.

**Exp. 2: Hierarchical classification.** We train a hierarchical cascade classifier ($SubCat_{hier}$), where a category classifier routes each sample to a dedicated per-category subcategory classifier. We additionally evaluate an oracle variant ($SubCat_{hier,orac}$) that uses ground-truth category labels for routing, establishing the upper bound of the hierarchical approach.

**Exp. 3: Cross-dataset evaluation.** To investigate whether aggregating multiple training sources under UCS improves generalization, we train a model on the combined EnvSound-UCS training set and evaluate it on each source's individual test set.

### 4.2. Classifier models and parametrization

Audio features are extracted using PANNs CNN14 [17], a pretrained convolutional neural network that yields 2048-dimensional embeddings for each audio clip. All classifiers share an identical architecture: a single linear layer projecting the 2048-dimensional embedding directly to the output classes, preceded by dropout ($p = 0.3$). This uniform design ensures that performance differences across experiments reflect the classification strategy rather than architectural variation. The setup is intentionally simple to establish an easily reproducible baseline.

All classifiers are trained with AdamW (learning rate $10^{-3}$, weight decay $10^{-3}$), cosine annealing learning rate scheduling, early stopping with patience of 20 epochs, and a maximum of 200 epochs. To address class imbalance, we use focal loss ($\gamma = 2.0$) with inverse-frequency class weighting. Identical hyperparameters are used for all datasets. Each experiment is repeated with five random seeds (42, 123, 456, 789, 1024) and we report mean ± standard deviation. The primary evaluation metric is the macro-averaged F1 score, which weights all classes equally regardless of their sample count and thus reflects performance on rare categories.

Note that PANNs CNN14 is pretrained on the full AudioSet training corpus (∼2M clips). For the AudioSet-UCS and EnvSound-UCS benchmarks, a portion of the test files originate from AudioSet's balanced training subset and were therefore seen during PANNs pretraining. While the UCS category labels used here differ from AudioSet's original 527-class labels, the learned representations may still benefit from prior exposure to these audio files. No part of FSD50K and ESC-50 were part of the PANNs training data.

**Table 4:** Macro F1, mean ± std (5 seeds) on environmental sound dataset classification with `MUSICAL` excluded and `VOICES` restricted to `CRYING` and `LAUGH`. Header notation (C/S) indicates the number of categories C and subcategories S. Upper section: self-trained models. Lower section: model trained on the combined EnvSound-UCS corpus, evaluated on each source's test set.

| | FSD-env (36/50) | AS-env (59/142) | ESC-50 (26/20) | EnvSound (59/144) |
|---|---|---|---|---|
| *Self-trained* | | | | |
| $Cat$ | $.53_{\pm.00}$ | $.47_{\pm.00}$ | $.89_{\pm.00}$ | $.46_{\pm.00}$ |
| $Cat_{flat}$ | $.72_{\pm.00}$ | $.57_{\pm.00}$ | $.99_{\pm.00}$ | $.55_{\pm.00}$ |
| $SubCat_{flat}$ | $.66_{\pm.00}$ | $.49_{\pm.01}$ | $.95_{\pm.01}$ | $.49_{\pm.00}$ |
| $SubCat_{hier}$ | $.61_{\pm.00}$ | $.42_{\pm.01}$ | $.95_{\pm.01}$ | $.39_{\pm.00}$ |
| $SubCat_{hier,orac}$ | $.88_{\pm.00}$ | $.74_{\pm.02}$ | $.95_{\pm.01}$ | $.73_{\pm.01}$ |
| *Trained on EnvSound-UCS* | | | | |
| $SubCat_{flat}$ | $.62_{\pm.00}$ | $.46_{\pm.00}$ | $.92_{\pm.00}$ | |

### 4.3. Results and Discussion

#### *4.3.1. Exp. 1: Flat classification*

The flat classification results $Cat$ are presented in Tables 3 and 4. ESC-50 (26 categories) can be classified with a F1 of .89, compared to .52 for FSD50K (37 categories) and .42 for AudioSet (60 categories). Performance scales inversely with the number of classes; this relationship is expected, as classifying 60 categories is inherently harder than 26, and must be considered when comparing results across datasets. The flat subcategory classification $SubCat_{flat}$ yields, interestingly, higher macro F1 across all datasets. Removing the `MUSICAL` category (Table 4, upper section) has only a negligible effect on performance, likely because `MUSICAL` is well-separated and acoustically distinct from the remaining categories. Across all datasets, $Cat_{flat}$—the category-level F1 derived from subcategory predictions—consistently exceeds the direct category classifier $Cat$ (e.g., .71 vs. .52 on FSD50K, .56 vs. .42 on AudioSet). This suggests that the top-level UCS categories are broad and internally heterogeneous, making them difficult to model directly with a linear classifier. Learning at the finer subcategory granularity, where classes can be assumed to be more acoustically homogeneous, and then aggregating to categories yields better category-level discrimination.

#### *4.3.2. Exp. 2: Hierarchical classification*

On all datasets, the flat subcategory classifier outperforms the cascaded approach: $SubCat_{hier}$ trails $SubCat_{flat}$ by 5 percentage points on FSD50K, 7 on AudioSet, and 10 on EnvSound-UCS. ESC-50 shows virtually no difference between the two approaches. This comparably poor performance is explained by the insufficient accuracy of the current category classifiers, which achieve F1 scores of only .42–.52 on the larger datasets. At these accuracy levels, routing errors propagate and impact the subcategory stage, outweighing the search space reduction benefits of hierarchical classification.

However, oracle routing reveals the potential of utilizing the hierarchy: $SubCat_{hier,orac}$ outperforms the flat baseline by +20 percentage points on FSD50K, +25 on AudioSet-env, and +24 on EnvSound-UCS. The result for ESC-50 exhibits no meaningful ora-

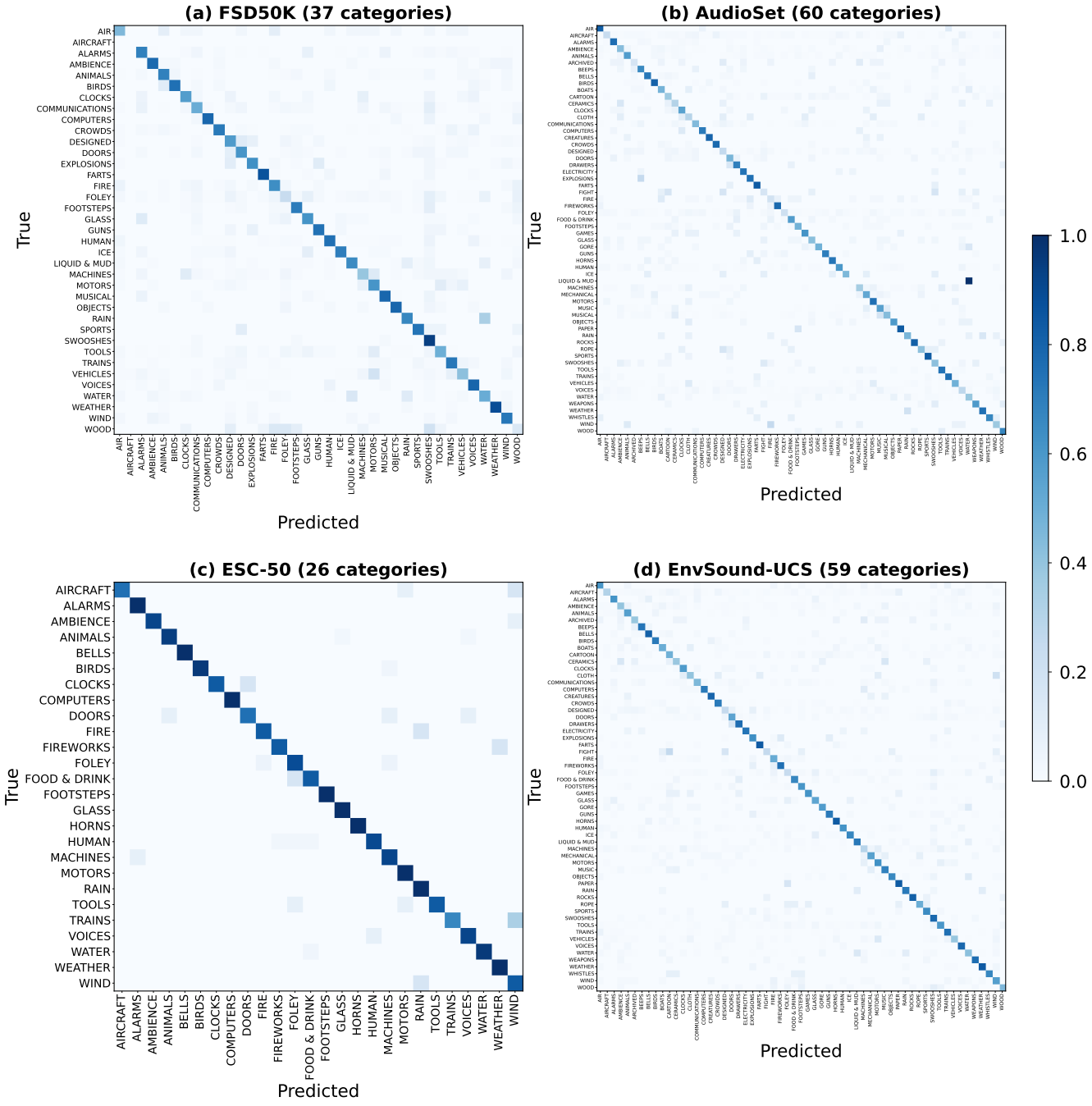


**Figure 3:** Row-normalized category confusion matrices for (a) FSD50K, (b) AudioSet, (c) ESC-50, and (d) EnvSound-UCS.

cle improvement, likely because the flat classifier already achieves a high F1 of 0.95 and the limited number of subcategories (20) does not benefit from hierarchical routing. This significant increase on the larger datasets indicates that improving category classifier accuracy could be key to unlocking the cascade's potential.

Figure 3 shows the category-level confusion matrices. Recurring off-diagonal confusion patterns include VEHICLES↔MOTORS and LIQUID & MUD↔WATER. These misclassifications follow acoustically motivated patterns: VEHICLES and MOTORS share engine-related sounds, making them inherently ambiguous at the category level, and LIQUID & MUD and WATER both involve fluid dynamics, potentially having similar characteristics.

#### 4.3.3. *Exp. 3: Cross-source evaluation*

The lower section of Table 4 reports the cross-source results. When the EnvSound-UCS model is evaluated on each source's test set, it

trails the corresponding self-trained model by 4 percentage points on FSD50K-env, 3 on ESC-50, and 3 on AudioSet-env.

This performance degradation raises the question of why more training data fails to improve results. A per-category analysis reveals that the primary cause is not class imbalance: the correlation between training distribution shift and per-category F1 change is not statistically significant for FSD50K-env (Pearson $r = -0.26$, $p = 0.13$), although a weak negative correlation is observed for AudioSet-env ($r = -0.35$, $p < 0.01$).

Instead, the following factors might explain the degradation. First, the number of training classes increases when sources are combined: FSD50K-env has 36 categories, but the EnvSound-UCS model must classify into 59—including 23 categories absent from FSD50K. This expanded output space increases training complexity without being directly relevant during inference. AudioSet-env (59 categories) experiences minimal class expansion (59→59), consistent with its smaller performance drop of only 3%.

Second, dataset homogeneity might be a problem where the curated data especially of smaller datasets like ESC-50 may not always reflect the variety of input sources of real-world settings. AudioSet, on the other hand, contains in-the-wild YouTube clips with variable recording conditions. When these heterogeneous sources are mixed, the model encounters less homogeneous acoustic representations for the same category. The ESC-50 category `GLASS` illustrates this effect: the per-category F1 drops from .92 (self-trained) to .25 (EnvSound-trained), suggesting that the clean, curated ESC-50 class might be more challenging to model when including noisier representations from other datasets.

Finally, there is a chance that data and annotation quality might impact results. AudioSet, in particular, is known to include quality-impaired audio samples and potentially noisy annotations [18]. Adding noisy labels or audio data to the training set might thus negatively impact inference performance.

However, the evidence is inconclusive as to whether a more sophisticated classifier or re-training the feature extractor from scratch might benefit from the larger training dataset and yield better results with respect to generalizability and accuracy.

## 5. LICENSING CONSIDERATIONS

Combining multiple datasets under a unified taxonomy raises licensing questions worth considering. We distinguish between *audio* licenses (governing the sound files themselves) and *metadata* licenses (governing annotations, ontologies, and derived labels).

**Audio licenses.** The three source datasets carry different audio licenses. FSD50K audio clips are individually licensed under various Creative Commons terms (CC0, CC-BY, CC-BY-NC, or CC Sampling+), with per-clip information available in the dataset metadata. ESC-50 audio is distributed under CC-BY-NC 3.0, prohibiting commercial use. AudioSet does not distribute audio directly; it provides YouTube video identifiers and time segments, with the original audio subject to each video's individual copyright.

**Metadata licenses.** FSD50K annotations are released under CC-BY 4.0. AudioSet annotations are CC-BY 4.0, but the AudioSet ontology (class hierarchy and labels) is CC-BY-SA 4.0, meaning derivative works must be released under the same license. Since our tag-to-UCS mapping table derives from the AudioSet ontology, it may be subject to the SA (ShareAlike) condition. The UCS taxonomy itself is public domain, imposing no restrictions. Our derived metadata (UCS labels, split files, and mapping tables) are released under CC-BY-SA 4.0.

**Implications for EnvSound-UCS.** Because EnvSound-UCS aggregates sources with different licenses, users who assemble the full audio corpus must respect the most restrictive terms: ESC-50 audio carries a non-commercial (NC) constraint, and AudioSet audio cannot be redistributed due to YouTube Terms of Service. Our framework distributes only CSV metadata files (UCS labels, splits, and conversion code), not audio files for any dataset, which mitigates redistribution concerns. Users must independently obtain audio from the original sources: Freesound/Zenodo for FSD50K, YouTube for AudioSet, and GitHub for ESC-50. Full per-dataset licensing details, together with composition, collection, and intended-use documentation, are provided in the per-dataset datasheets [4].[2]

## 6. CONCLUSION

We have presented a UCS-based framework for standardized conversion and hierarchical annotation of sound effects datasets. The CSV-based tag-to-UCS conversion pipeline achieves automatic mapping rates of 100% on FSD50K and ESC-50, and 98.49% on AudioSet. The UCS-aware re-splitting tool preserves category and subcategory distributions through composite stratification keys and supports multi-source merging, as demonstrated by the construction of a 58,057-sample environmental sound corpus from three sources after category filtering (Section 3.1). Beyond aggregation, the framework's category-level filtering enables the extraction of targeted subsets (e.g., a focused outdoor ambience corpus from `WEATHER`, `WIND`, and `RAIN` categories), addressing the core goals outlined in the introduction: compatibility and interoperability through a shared label space, extensibility through a reusable conversion pipeline, and coherence through the hierarchical UCS structure that supports both broad and focused dataset construction.

The benchmark experiments reveal three key findings. First, category-level F1 derived from subcategory predictions consistently exceeds direct category classification, suggesting that the top-level UCS categories are broad and internally heterogeneous, making finer-grained subcategory learning a more effective strategy even for coarse-level tasks. Second, oracle routing yields +20–25% improvements over flat baselines on the larger datasets, demonstrating the latent potential of hierarchical classification under accurate category routing. Third, naive aggregation of heterogeneous data sources does not automatically improve over single-source training.

The deliverables of this work are distributed across six public GitHub repositories (Section 1). The tool repositories contain the conversion pipeline, re-splitting tool, and configuration files; the data repositories contain metadata only (per-file UCS labels, train/validation/test partitions, and ambiguity review lists). Audio files are not redistributed. Datasheets following Gebru et al. [4] are provided for all four datasets in a dedicated documentation repository. By bringing an industry-standard taxonomy into the academic domain, this work bridges the gap between professional sound design practice and audio machine learning research. With this release we hope to facilitate future research focusing on environmental sounds that links academic research with industry needs.

### 6.1. Limitations and Future Work

Several limitations should be noted. First, the conversion pipeline operates entirely on text-based tag matching and does not perform auditory verification of the assigned categories. The resulting UCS

[2]Datasheet repository:
`https://github.com/JunWooBeck/ucs-sfx-datasheets`

labels inherit any noisy labels in the original source annotations. Second, the positional priority tie-breaking rule is invoked for approximately 10–20% of classified files across datasets. All files where tie-breaking was applied are recorded in the ambiguity review list, enabling users to identify and manually verify these cases. Third, UCS version updates may require re-mapping; all results reported here are based on UCS v8.2.1. Fourth, the benchmark experiments use PANNs CNN14 embeddings pretrained on the full AudioSet training corpus [17] ($\sim$2M clips). For AudioSet-UCS, approximately 51% of test files originate from AudioSet's balanced training subset and were therefore seen during PANNs pretraining; for EnvSound-UCS this figure is approximately 20%. While the UCS category labels used here differ from AudioSet's original 527-class labels, the learned representations may still benefit from prior exposure to these audio files. FSD50K and ESC-50 are not part of the PANNs training data and are unaffected. Finally, the current pipeline requires a manually curated mapping table to handle dataset-specific tag conventions. While this table is reusable across subsequent conversions of the same dataset, extending the framework to new datasets requires adding new mappings. Future work could leverage large language models to automate this mapping process, and domain-aware training strategies (e.g., source-weighted sampling or domain adaptation) could address the performance degradation observed in naive multi-source aggregation. The potential of hierarchical classification demonstrated by the oracle experiments could be further explored through taxonomy-aware learning [19] and multimodal approaches [20].

## 7. ACKNOWLEDGMENTS

The authors thank the anonymous reviewers for their feedback and the UCS community for maintaining the taxonomy as a public resource.

## 8. REFERENCES

[1] L. Liu, V. Morfi, and E. Benetos, "ACPAS: A dataset of aligned classical piano audio and scores for audio-to-score transcription," in *Late-Breaking Demo, Proc. Int. Society for Music Information Retrieval Conf. (ISMIR)*, 2021.

[2] R. M. Bittner, M. Fuentes, D. Rubinstein, A. Jansson, K. Choi, and T. Kell, "mirdata: Software for reproducible usage of datasets," in *Proc. Int. Society for Music Information Retrieval Conf. (ISMIR)*, 2019.

[3] A. B. Ma and A. Lerch, "Representation learning for the automatic indexing of sound effects libraries," in *Proc. Int. Society for Music Information Retrieval Conf. (ISMIR)*, Bengaluru, India, 2022, pp. 866–875.

[4] T. Gebru, J. Morgenstern, B. Vecchione, J. W. Vaughan, H. Wallach, H. Daume III, and K. Crawford, "Datasheets for datasets," *Communications of the ACM*, vol. 64, no. 12, pp. 86–92, Dec. 2021.

[5] D. Moffat, D. Ronan, and J. D. Reiss, "Unsupervised taxonomy of sound effects," in *Proc. Int. Conf. Digital Audio Effects (DAFx-17)*, Edinburgh, UK, 2017.

[6] D. Sonnenschein, *Sound Design: The Expressive Power of Music, Voice, and Sound Effects in Cinema*. Studio City, CA: Michael Wiese Productions, 2001.

[7] International Phonetic Association, *Handbook of the International Phonetic Association: A Guide to the Use of the International Phonetic Alphabet*. Cambridge, U.K.: Cambridge University Press, 1999.

[8] "Universal Category System (UCS) for sound effects, version 8.2.1," [Online], Mar. 2026, available: https://universalcategorysystem.com.

[9] P. Cano, M. Koppenberger, O. Celma, P. Herrera, and V. Tarasov, "Sound effects taxonomy management in production environments," in *Proc. AES 25th Int. Conf.*, London, UK, 2004.

[10] E. Fonseca, X. Favory, J. Pons, F. Font, and X. Serra, "FSD50K: An open dataset of human-labeled sound events," *IEEE/ACM Trans. Audio, Speech, and Language Processing*, vol. 30, pp. 829–852, 2022.

[11] 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 *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, New Orleans, LA, 2017, pp. 776–780.

[12] K. J. Piczak, "ESC: Dataset for environmental sound classification," in *Proc. ACM Int. Conf. Multimedia*, Brisbane, Australia, 2015, pp. 1015–1018.

[13] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proc. ACM Int. Conf. Multimedia*, Orlando, FL, 2014, pp. 1041–1044.

[14] K. Drossos, S. Lipping, and T. Virtanen, "Clotho: An audio captioning dataset," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 736–740.

[15] P. Stamatiadis, M. Olvera, and S. Essid, "SALT: Standardized audio event label taxonomy," in *Proc. Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, Tokyo, Japan, 2024.

[16] P. Anastasopoulou, J. Torrey, X. Serra, and F. Font, "Heterogeneous sound classification with the Broad Sound Taxonomy and dataset," in *Proc. Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, Tokyo, Japan, 2024, pp. 11–15.

[17] 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 Trans. Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[18] 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.

[19] J. Liang, H. Phan, and E. Benetos, "Learning from taxonomy: Multi-label few-shot classification for everyday sound recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, Seoul, Korea, 2024, pp. 771–775.

[20] P. Anastasopoulou, F. Dal Rí, X. Serra, and F. Font, "Hierarchical and multimodal learning for heterogeneous sound classification," in *Proc. Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, Barcelona, Spain, 2025, pp. 105–109.