Audio Classification - Data layer

Datasets — Benchmarks & Sources

ESC-50

- **What it is:** 2,000 environmental audio clips (5 s) across 50 classes (animals, natural, human, interior, exterior).
- **Why it matters:** Clean, balanced, classic baseline for environmental sound classification.
- **Quirks:** Small size → risk of overfitting; 5 predefined cross-val folds.
- **Where:** Hugging Face (e.g., `ashraq/esc50`), original site (Piczak).

UrbanSound8K

- **What it is:** 8,732 urban sound clips (≤4 s) in 10 classes (siren, drill, dog bark, etc.).
- **Why it matters:** Standard for noisy, real-world urban acoustics; fold splits provided.
- **Quirks:** Variable durations; city noise; strong domain shift to other locales.
- **Where:** Hugging Face (`urbansound8k`), original site (urbansounddataset.weebly).

AudioSet (Balanced / Unbalanced)

- **What it is:** >2M 10-s YouTube clips weakly labeled with 527 classes (ontology by Google).
- **Why it matters:** The go-to large-scale, multi-label benchmark; powers many SOTA models.
- **Quirks:** Weak/noisy labels; long-tail class imbalance; requires YouTube downloads (availability drift).
- **Where:** Google Research (ontology + CSVs + embeddings); HF Hub hosts community shards/embeddings (search “audioset”).

FSD50K

- **What it is:** ~51k Freesound clips labeled with AudioSet ontology (multi-label).
- **Why it matters:** Curated alternative to AudioSet with improved labels; good for pretraining/fine-tuning.
- **Quirks:** Multi-label; class imbalance; varied clip lengths.
- **Where:** Zenodo (official); Hugging Face community mirrors (search “FSD50K”).

DCASE Acoustic Scenes (e.g., TAU Urban Acoustic Scenes 2019/2020)

- **What it is:** 10-s clips from multiple cities labeled with acoustic scenes (park, metro, etc.).
- **Why it matters:** Benchmark for acoustic scene classification with rigorous challenge splits.
- **Quirks:** Strong location/device domain shifts in some tracks; device mismatch protocols.
- **Where:** DCASE challenge pages; some subsets mirrored on HF (search “TAU Urban”).

Speech Commands v2 (Google)

- **What it is:** ~105k 1-s utterances of keywords (yes/no/up/down/…).
- **Why it matters:** De-facto benchmark for small-footprint keyword spotting.
- **Quirks:** Background noise fold; speaker imbalance; 1-s fixed window.
- **Where:** Hugging Face (`speech_commands`), TensorFlow datasets, Google.

GTZAN (Music Genre)

- **What it is:** 1,000 30-s music clips across 10 genres.
- **Why it matters:** Historical genre benchmark; quick baselines.
- **Quirks:** Known label/partition issues; potential artist leakage—use for teaching/prototyping only.
- **Where:** HF (`gtzan`), original mirrors.

MagnaTagATune / MTG-Jamendo (Music Tagging)

- **What it is:** Music tagging datasets with multiple labels (instruments, mood, genre).
- **Why it matters:** Standard for multi-label music tagging and transfer to downstream tasks.
- **Quirks:** Tag imbalance; artist leakage concerns—use artist-conditional splits.
- **Where:** HF (`magnatagatune`, `mtg_jamendo_*` collections), original sites.

NSynth

- **What it is:** ~300k musical notes from 1k+ instruments (pitch/timbre attributes).
- **Why it matters:** Instrument/timbre classification; good for controlled audio.
- **Quirks:** Synthetic/isolated notes may not generalize to real mixes.
- **Where:** HF (`nsynth`), Magenta (Google).

VoxCeleb1/2 (Speaker ID)

- **What it is:** Large-scale speaker identification from YouTube interviews.
- **Why it matters:** Speaker classification/verification baselines; robust to real-world noise.
- **Quirks:** Label noise; overlapping backgrounds; long-tail speakers.
- **Where:** Official site; HF mirrors (search “voxceleb”).

SUPERB (Benchmark Suite)

- **What it is:** A unified suite covering many speech tasks, incl. keyword spotting and intent classification.
- **Why it matters:** Consistent evaluation across models; easy HF integration.
- **Quirks:** Mixed tasks and metrics; ensure you isolate classification tasks.
- **Where:** Hugging Face (`superb`).

Preprocessing (what to do and why)

Resampling & Channeling

We bring all audio to a consistent sample rate and channel layout for stable training and batching.

- **Resample to 16 kHz or 32 kHz (mono):** Standardizes time resolution and reduces compute while matching many pretrained models' expectations.
- **Convert to mono (if appropriate):** Removes channel variance; most benchmarks are single-channel.

Loudness & Gain Normalization

We normalize levels so models don't learn trivial volume cues.

- **Peak/RMS/LUFS normalization:** Keeps input dynamic range consistent across recordings and devices.

Trimming & Silence Handling

We remove leading/trailing silence and optionally pad to a target length for uniform batches.

- **Trim silence, then pad/clip to fixed window (e.g., 1 s/5 s/10 s):** Reduces wasted compute; aligns to dataset clip lengths (Speech Commands = 1 s, ESC-50 = 5 s, DCASE = 10 s).

Time–Frequency Features

We convert waveforms to features that models can learn from efficiently.

- **Log-mel spectrograms (e.g., n_fft=1024, hop=10 ms, win=25 ms, n_mels=64–128):** Strong baseline for CNN/AST models.
- **MFCCs (e.g., 13–40 coeffs + deltas):** Lightweight classic features for small models or on-device KWS.

Normalization

We adjust feature values so they're centered and scaled, making training stable.

- **Per-feature standardization (mean/var over train set):** Zero-mean unit-var for spectrogram bins.
- **Per-utterance CMVN (cepstral mean/variance normalization):** Useful when recording conditions vary widely.

Augmentation

We diversify data to improve robustness and generalization.

- **SpecAugment (time/freq masking):** Regularizes time-freq features without external noise.

- **Time stretch / pitch shift (small factors):** Simulates tempo/pitch variability in speech/music.
- **Background noise mixing (e.g., MUSAN):** Improves noise robustness for KWS/ASC.
- **Random gain / reverb / bandpass:** Matches diverse microphones and rooms.

Multi-label Handling (where applicable)

We adapt labels and losses for datasets with multiple tags per clip.

- **Binary indicator vectors + sigmoid + BCE loss:** Needed for AudioSet/FSD50K/Music tagging.

Dataset-Specific Notes

- **AudioSet/FSD50K:** Multi-label, long-tail → use class-balanced sampling or focal loss; consider weak-label MIL pooling.
- **ESC-50/UrbanSound8K:** Use official folds; 5 s/≤4 s windows; stratified evaluation.
- **Speech Commands:** Fixed 1 s window; include `_background_noise_` for realism.
- **DCASE:** Watch device/domain splits; train with device augmentation or domain adaptation.
- **GTZAN/MagnaTagATune/Jamendo:** Beware artist/album leakage; use artist-conditional splits.

Dataloading tips

We prepare the dataset so training is fast, reproducible, and efficient.

- **Prefetch & pin memory:** `DataLoader(pin_memory=True, prefetch_factor=2)` → Keeps GPUs fed while decoding spectrograms.
- **Worker init functions:** `worker_init_fn=seed_all` → Ensures random crops/masks are reproducible.
- **On-the-fly feature extraction:** Cache log-mels to disk (`dataset.with_transform(...)`) → Avoids recomputing spectrograms every epoch.
- **Balanced sampling:** Class-aware sampler or reweighting for long-tail sets → Prevents majority classes from dominating.
- **Deterministic validation:** Fixed crop or full-clip evaluation; no augmentation → Fair comparisons across checkpoints.
