Offline Soundscape Validation and Rank-Blended CNN Ensembles for BirdCLEF+ 2026

Gilles Colling[1,2]

[1]Division of BioInvasions, Global Change & Macroecology, Department of Botany and Biodiversity Research, University of Vienna, Vienna, Austria
[2]Vienna Doctoral School of Ecology and Evolution, University of Vienna, Vienna, Austria

Abstract

BirdCLEF+ 2026 evaluates multi-taxon acoustic species identification across 234 classes spanning birds, amphibians, insects, and reptiles, scored by macro-averaged ROC-AUC on a hidden soundscape test set whose public leaderboard is computed on only part of that set during the competition. A single competitor sees a few public-leaderboard readings per day, which raises two practical questions: how to evaluate candidate systems without spending that budget, and how far a public score predicts the private one. We report a single-author entry, built on public resources, that addresses both. The submitted system rank-blends a publicly shared acoustic baseline (a Perch-embedding branch and a five-fold sound-event-detection ensemble) with a weighted triplet of our own soundscape-fine-tuned convolutional networks at weight $\alpha = 0.10$. To rank configurations offline we built a harness on the 66 released labelled soundscape files, with macro-AUC restricted to the 75 species that carry positive labels there; it scores a blend in under a second and replaced the daily leaderboard slot as the primary feedback channel. The entry scored 0.932 macro-AUC on the public leaderboard (1968th of 4092 teams) and 0.917 on the private leaderboard (2415th), and the soundscape-tuned correction helped only at a low blend weight. The two leaderboards did not rank our submissions in the same order, and the harness could not have predicted this: the dominant embedding branch trains on the same labelled soundscapes, so no clean held-out estimate of it exists. We report this generalisation gap and identify a non-leaky cross-validated estimate of the full blend as the change most likely to have closed it.

Keywords

bird sound classification, soundscape, sound event detection, pseudo-labelling, cross-year pretraining, model ensembling, macro-averaged ROC-AUC, offline model validation, leaderboard generalisation gap, BirdCLEF, EfficientNet, Perch

1. Introduction

BirdCLEF+ 2026 asks systems to identify which species vocalise in a continuous field recording. The task is framed as multi-label detection over 5-second windows: for each window of a hidden soundscape recording, a submission reports a probability for each of 234 candidate classes, and the predictions are scored by macro-averaged ROC-AUC over the classes that have positive labels in the test set [4, 5, 3]. The 2026 edition is multi-taxon: the 234 classes span birds, amphibians, insects, and reptiles, so the same model must separate species with very different call structures. Two properties shape the design of any entry. First, training labels are clip-level and weakly localised, while scoring is window-level on soundscapes, a distribution shift between training and evaluation audio. Second, inference runs on a CPU-only Kaggle kernel under a wall-clock budget, which caps how large and how numerous the models in an ensemble can be.

We entered BirdCLEF+ 2026 as a single author, building on public components. The submitted system is an ensemble of two parts. The first part is a publicly shared acoustic baseline that we did not author: an embedding branch built on the Perch bird-vocalisation model [7] with a probe-classifier head, and a five-fold EfficientNet sound-event-detection ensemble [9]. The second part is our own contribution: a weighted triplet of convolutional networks fine-tuned on the released soundscape recordings. The two

CLEF 2026: Conference and Labs of the Evaluation Forum, September 21–24, 2026, Jena, Germany
gilles.colling051@gmail.com (G. Colling)
0000-0003-3070-6066 (G. Colling)

parts are combined by rank-blending the soundscape-tuned triplet onto the public baseline at a small weight $\alpha = 0.10$, an operating point chosen on local validation and confirmed on the leaderboard.

A single competitor gets only a few public-leaderboard readings per day, and a public score estimates the private result with its own error. Two of the pieces we report respond to these limits. One is a local evaluation harness on the 66 labelled soundscape files released with the competition, with macro-AUC restricted to the 75 species that actually have positive labels there; it let us rank ensemble configurations offline instead of spending a daily leaderboard slot per hypothesis. The other is a development track, following recipes reported by prior BirdCLEF winning solutions, that added cross-year pretraining on previous editions and frame-level multi-hot pseudo-labels; that track was only partly completed before the deadline, and we report what was finished and what was not.

Our contributions are: (i) a local soundscape evaluation harness that replaced the daily leaderboard slot as the primary feedback channel, scoring any model or blend specification in under a second; (ii) an analysis of a public-to-private generalisation gap: the configuration with the highest public score was not the best on the private split, because the branch that dominates the blend trains on the same labelled soundscapes used for validation and so cannot be held out cleanly; (iii) the submitted ensemble itself, a public Perch and sound-event-detection baseline blended with a weighted triplet of soundscape-fine-tuned convolutional networks at $\alpha = 0.10$, with a component ablation that locates its score (0.932 macro-AUC on the public leaderboard, 0.917 on the private); and (iv) a cross-year-pretraining and multi-hot pseudo-labelling development track, with an honest account of which parts were completed and of the inference-budget constraints that bounded the model zoo.

2. Related Work

2.1. BirdCLEF and prior winning recipes

BirdCLEF has run annually within the LifeCLEF lab and has converged on a recognisable recipe [6, 5, 4]. Across recent editions, the strongest solutions share several elements: log-mel spectrograms over short windows with sliding-window test-time augmentation; convolutional backbones from the EfficientNet [9] and ConvNeXt [10] families trained with waveform and spectrogram augmentation; pretraining on prior-year BirdCLEF audio and on Xeno-canto recordings followed by fine-tuning on the current year; and one or more rounds of pseudo-labelling on unlabelled soundscapes. Recent editions added frame-level multi-label pseudo-targets in place of clip-level argmax labels, and a pairwise AUC-surrogate loss in the final training stage. We adopt several of these elements in the development track of Section 8; the soundscape-tuned models in the submitted system use the window-and-augmentation part of this recipe without the cross-year pretraining.

2.2. Acoustic embeddings and sound event detection

Two model families recur in bird-audio systems. Pretrained acoustic embedding models, such as BirdNET [8] and Perch [7], are trained on large labelled bird-audio corpora and expose either direct class probabilities or transferable embeddings that a lightweight head can probe. Sound-event-detection (SED) models instead train a convolutional encoder with an attention or log-sum-exp pooling head that aggregates frame-level activations into clip-level scores, which suits weakly labelled training audio. The public baseline we build on combines both: a Perch-embedding branch with a probe head, and an EfficientNet SED ensemble. The soundscape-tuned models we add are SED-style convolutional networks.

2.3. Pseudo-labelling for soundscapes

Training audio in BirdCLEF is dominated by focal recordings of a single vocalising individual, whereas the test audio is continuous soundscape with overlapping species and long silent stretches. Pseudo-labelling closes part of this gap: a teacher model labels unlabelled or weakly labelled windows, and

a student is trained on the resulting targets. A single-class argmax target discards the multi-species structure of real soundscapes; a frame-level multi-hot target preserves it. Section 8 reports a multi-hot pseudo-labelling pass over the soundscape, training, and prior-year corpora.

2.4. The evaluation metric

Submissions are scored by macro-averaged ROC-AUC: the per-class ROC-AUC is computed against the window-level binary ground truth and averaged over classes, skipping any class with no positive label in the test set [14]. Two consequences shape both training and validation. Because the average is taken over classes, a handful of rare species can move the score as much as a common one. And because classes without positives are skipped, a validation set that contains positives for only a subset of the 234 classes must restrict its macro-average to that subset, or the score is dominated by uninformative chance-level entries; we return to this in Section 7.

3. Task and Data

The released competition data comprise clip-level training audio, a small labelled soundscape set, and a hidden soundscape test set [1, 2]. The provided files are:

- `train_audio` – 35 549 focal recordings across 206 species directories, with clip-level primary and secondary labels in `train.csv`;
- `train_soundscapes` – 66 continuous soundscape recordings with window-level labels in `train_soundscapes_labels.csv` (1 478 labelled 5-second windows covering 75 of the 234 classes);
- `taxonomy.csv` – the 234 target classes with iNaturalist taxon identifiers and class names, spanning birds, amphibians, insects, and reptiles.

Submissions assign a probability to each of the 234 classes for every 5-second window of every test recording. Public-leaderboard scores are computed on a fraction of the hidden test set during the competition; private (final) scores are computed on the complement and revealed at the end. Throughout this paper we report both: public-leaderboard macro-AUC for development comparisons, and private-leaderboard macro-AUC where it is available for a submission.

Inference budget. Inference runs in a Kaggle notebook with no internet access and a CPU-only runtime cap on the hidden test set. This budget is a hard design constraint: it sets how many models can take part in the ensemble and rules out backbones whose per-window latency does not fit. Section 10 reports the conversions and the one heavy model the budget forced us to drop.

4. System Overview

Figure 1 sketches the submitted pipeline. It has two parts that are combined at the very end. The public baseline produces a per-window score from a Perch-embedding probe branch (weight 0.7) and a five-fold SED ensemble (weight 0.3), rank-blended together. Our contribution is a triplet of soundscape-tuned convolutional networks, combined by a weighted rank-mean. The triplet is then rank-blended onto the baseline at $\alpha = 0.10$: the baseline carries most of the final score and the triplet supplies a small, consistent correction. We use rank-blending rather than a weighted probability average because the component models are not calibrated to a common scale, and rank-blending is insensitive to that mismatch.

All models share a common front-end. Audio is resampled to 32 kHz mono and cut into 5-second windows (twelve windows per 60-second soundscape recording). Each window is converted to a 128-band log-mel spectrogram with a 1024-point FFT, 320-sample hop, and a 50–14 000 Hz mel range, peak-normalised per window. The convolutional models take this single-channel spectrogram as input.

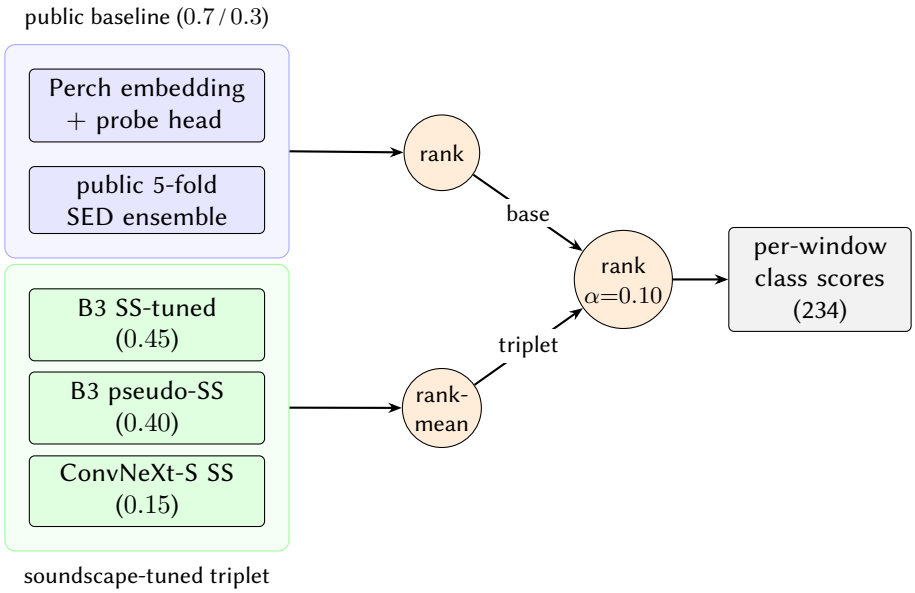

Figure 1: Submitted system. A publicly shared acoustic baseline (a Perch-embedding probe branch and a five-fold SED ensemble, rank-blended $0.7 / 0.3$) is combined with our own weighted triplet of soundscape-tuned convolutional networks (rank-mean of an EfficientNet-B3, a pseudo-refined B3, and a ConvNeXt-Small at $0.45 / 0.40 / 0.15$). The triplet is rank-blended onto the baseline at $\alpha = 0.10$ to produce the per-window class scores.

5. Soundscape-Tuned Models

The triplet is the part of the submitted system we trained ourselves. Each member is a convolutional SED model first trained on the clip-level `train_audio` labels and then fine-tuned on the 66 labelled soundscape recordings, which adapts it to the soundscape distribution that scoring uses.

Backbones. The triplet members are an EfficientNet-B3 [9] (`tf_efficientnet_b3.ns_jft_in1k`), the same B3 architecture fine-tuned with an extra pseudo-label pass on soundscape windows, and a ConvNeXt-Small [10] (`convnext_small.fb_in22k_ft_in1k`); all are taken from the `timm` library [13] with a single-channel input stem. The clip-level stage uses an asymmetric loss [11] over the multi-label targets. The soundscape fine-tuning stage trains on the labelled soundscape windows.

Triplet weights. The members are combined by a rank-mean with weights 0.45 (B3), 0.40 (pseudo-refined B3), and 0.15 (ConvNeXt-Small). These weights were tuned on the local validation harness of Section 7 rather than on the leaderboard. An equal-weight rank-mean of the same three members scored within 0.0001 of the weighted triplet on the public leaderboard (Table 3), so the weighting is a small effect; we report it for reproducibility.

6. The Public Baseline Branch

The baseline branch is a publicly shared community pipeline that we adopted and did not modify, and we describe it here for completeness and to be clear about provenance. It has two components. The first is an embedding branch built on the Perch bird-vocalisation model [7]: Perch embeddings feed a probe-classifier ensemble whose outputs are corrected by a small state-space model, contributing a weight of 0.7 to the baseline. The second is a five-fold EfficientNet-B0 SED ensemble [9] trained with log-sum-exp pooling and pseudo-labels, contributing a weight of 0.3. The two are rank-blended to form the baseline prediction onto which our triplet is added.

Provenance. For reproducibility we record the exact public artifacts. The baseline was adopted from the publicly shared Kaggle notebook `needless090/birdclef-2026-iter-pseudo-perch-sed-lb-0-935-ft` (public-leaderboard score 0.935), which we forked without modifying its Perch or SED logic; our submitted kernel is `gillescolling/blend-v2-b0`. The embedding branch uses the Google Perch v2 vocalisation classifier (Kaggle model `google/bird-vocalization-classifier/TensorFlow2/perch_v2_cpu/1`; [7]), served for CPU inference through the community datasets `needless090/birdclef2026-perch-tflite`, `jaejohn/perch-meta`, and `rishikeshjani/perch-onnx-for-birdclef-2026`. The five-fold EfficientNet-B0 SED ensemble is distributed as `hungtrabiz/birdclef-2026-submission-assets`. These community datasets and notebooks carry no embedded licence metadata on Kaggle; we use them unchanged and reference rather than redistribute them, so they remain under their original authors' terms, and the Perch model under Google's model terms.

We used the branch unchanged in the submitted system; a separate attempt to retrain its embedding head is reported in Section 8 and on the leaderboard in Table 3.

7. Local Evaluation Harness

For much of the competition, hypotheses were tested by submitting to the leaderboard, which limits feedback to a few configurations per day. The released soundscape labels make a local alternative possible. We built an evaluator over the 66 labelled soundscape files: it scores any model or blend specification, computes per-class ROC-AUC on the window-level labels, and reports the macro-average. With the soundscape predictions cached, it scores a blend configuration in under a second, so tens of configurations can be ranked in an evening. The harness needs only the released soundscape labels and cached per-window predictions, so any entrant can reproduce it without spending a leaderboard submission. Two design choices follow from the metric (Section 4).

First, only 75 of the 234 classes have any positive label in the soundscape set, so a naive macro-AUC over all 234 classes is dominated by the 159 classes pinned at chance level; the soundscape fine-tuning logs showed exactly this, with an apparent validation AUC near 0.53. The harness restricts the macro-average to the 75 labelled classes, which makes it discriminating. Second, the harness uses k-fold cross-validation across the 66 files rather than the small file-level holdout the fine-tuning code used by default, which reduces variance enough to read differences of a few thousandths.

Table 1
Local soundscape harness, macro-AUC restricted to the 75 labelled species. All rows are out-of-fold predictions under a file-level 5-fold split of the 66 soundscape files (GroupKFold by file), averaged over folds. *worst-3* is the mean of the three lowest per-class AUCs, the tail the leaderboard is most sensitive to. The three-way blend of the two soundscape-tuned members with the public SED branch is strongest on every column.

Configuration	macro-AUC (pos)	worst-3
B3 soundscape-tuned, alone	0.9965	0.9731
ConvNeXt-Small soundscape-tuned, alone	0.9956	0.9671
B3 + ConvNeXt-Small (70 / 30)	0.9975	0.9807
B3 + public SED (70 / 30)	0.9962	0.9761
B3 + ConvNeXt-Small + SED (50 / 25 / 25)	**0.9988**	**0.9909**

All rows of Table 1 are out-of-fold predictions under the file-level 5-fold split of Section 9: each soundscape file is scored only by the fold in which it is held out, so no file contributes to both training and evaluation of the soundscape-tuned and SED branches.

Table 1 reports the headline readings. On this surface the soundscape-tuned models already saturate near the top of the macro-average, so the discriminating signal is the worst-3 tail. Adding the public SED branch to the two soundscape-tuned members lifts worst-3 from 0.9807 to 0.9909, while adding

SED to the B3 alone is close to neutral. The harness has one important limitation: the embedding branch trains on the same 66 soundscape files in submission mode, so a held-out Perch reading on this surface is leaky and we use the harness to rank the soundscape-tuned and SED parts with the embedding branch held fixed, not to estimate the embedding branch in isolation.

The harness predicts *rankings*, not absolute scores. Its macro-average sits near 0.99 because it is restricted to the 75 classes with positive labels and computed on clean window-level annotations, whereas the public leaderboard near 0.93 averages over a larger, partly different class set on the hidden distribution. The two scales are therefore not comparable in absolute terms; the harness earns its keep by ordering configurations that differ by a few thousandths, and that ordering transferred to the leaderboard for the parts it can see (the triplet-plus-SED worst-3 gain in Table 1 corresponds to the small but repeatable $0.927 \rightarrow 0.932$ lift in Table 3). It did not transfer for the embedding branch, which it cannot hold out.

8. Development Track: Cross-Year Pretraining and Multi-Hot Pseudo-Labelling

The submitted system plateaued near 0.932 on the public leaderboard. Prior BirdCLEF winning solutions reach higher with a deeper data pipeline rather than a different blend, so we pursued that pipeline in parallel.

Cross-year corpus (completed). We assembled a pretraining corpus from the current and prior editions: the 2026 training audio (35 549 recordings) and a small extra set (1 020 recordings) merged with the full 2025 training set (28 564 recordings across 206 species, 7.23 GB), for 65 133 recordings in total. Of these, 50 288 carry labels mappable to the 234 target classes (covering 231 of them); the remaining prior-year recordings retain their own labels. The pretraining label space is the union of the 234 target classes and 163 unmapped prior-year classes (397 classes), on the principle that the unmapped audio still teaches transferable acoustic features.

Multi-hot pseudo-labels (completed). A teacher ensemble of the triplet members labelled the soundscape, training, and prior-year windows. The teacher produced per-class probabilities for 482 981 windows, stored as multi-hot targets at several thresholds. We selected a threshold of 0.7 for the first fine-tuning pass, which yields 315 087 positive (window, class) entries with at least one positive in 228 of the 234 classes, retaining multi-species structure in the soundscape windows (3.8 positives per window) while filtering teacher noise on the focal recordings. A second pseudo round, with the development models as teacher, produced 282 537 windows.

Pretrain-then-fine-tune zoo (partially completed). The trainer uses a dual-head architecture, a clip-level attention head and a max-segment head pooled over the strongest frames, with loss weights 1.0 and 0.5 and label smoothing 0.1, following the dual-objective recipe of prior solutions. The loss schedule runs dual binary cross-entropy for the bulk of training and switches to a pairwise AUC-surrogate loss for the final five epochs of each stage. Augmentation combines waveform mixup, spectrogram mixup and cutmix, SpecAugment-style time and frequency masking [12], and background-noise injection sampled from the soundscape recordings. Stage 1 pretrains on the 397-class corpus; Stage 2 fine-tunes at window granularity on the 234 classes with ground-truth labels unioned with the multi-hot pseudo-targets.

Of the four backbones planned for this zoo (EfficientNet-B3, EfficientNet-V2-S, ECA-NFNet-L0, ConvNeXt-Small), only the EfficientNet-B3 line and a ResNeSt fine-tune were carried to completion before the deadline. The B3 reached a pretraining macro-AUC of 0.964, and its soundscape-dominant fine-tune (using $116\,970$ soundscape windows in place of the original 783) reached a within-training macro-AUC of 0.974. This model was submitted both on its own and as a fourth anchor in the triplet (Table 3). All training ran on two personal machines: a desktop with a single NVIDIA RTX 5080 (16 GB) and an Apple M4 Pro Mac mini (Metal Performance Shaders, 64 GB unified memory), with no cloud,

cluster, or multi-GPU-per-model scaling. Each model was trained on one device at a time, so the training throughput of these two consumer GPUs within the competition window set how much of the zoo we could complete. The full multi-backbone, multi-seed zoo that prior winning solutions use was not completed, and the development track therefore did not reach the higher leaderboard range it was aimed at.

9. Implementation Details

Table 2 collects the training settings for the two regimes used in this work: the soundscape fine-tuning that produced the submitted triplet, and the cross-year pretrain-then-fine-tune of the development track. All models share the front-end of Section 4 (32 kHz mono, 5-second windows, 128-band log-mel with a 1024-point FFT, 320-sample hop, 50–14 000 Hz range, per-window peak normalisation). Backbones are initialised from the `timm` weights named in Section 5 with a single-channel input stem.

Table 2
Training settings. The left column is the soundscape fine-tuning that produced the submitted triplet (EfficientNet-B3, pseudo-refined B3, ConvNeXt-Small); the right column is the cross-year development track (Section 8). The pseudo-refined B3 adds a fine-tuning pass on the multi-hot pseudo-targets of Section 8.

	Soundscape fine-tune (triplet)	Cross-year pretrain + FT (dev)
Optimiser	AdamW, weight decay 0.01	AdamW, weight decay 0.01
Learning rate	3×10^{-4}	1×10^{-3}
LR schedule	cosine, 1-epoch warmup	cosine, 2-epoch warmup
Epochs	20	30 (Stage 1 pretrain)
Batch size	32	64
Loss	ASL (γ_+=0, γ_-=4, clip 0.05)	dual BCE (clip 1.0 / segment 0.5); pairwise AUC-surrogate in the final 5 epochs
Label smoothing	0.02	0.1
Augmentation	mixup (α=0.3, p=0.4); SpecAugment (2 time masks ≤ 24, 2 freq masks ≤ 16)	waveform mixup (p=0.5, α=0.4); spectrogram mixup and cutmix (p=0.25); SpecAugment (time 24, freq 16); soundscape-noise injection at window-level FT (Stage 2)
Soundscape oversampling	4–$6\times$	
Gradient clip	5.0	5.0
Seed	42	42
Validation split	5-fold by soundscape file (66 files, GroupKFold)	per-class stratified 5% holdout
Checkpoint	best soundscape macro-AUC	best macro-AUC

The multi-hot pseudo-labels were produced by an unweighted average of the three triplet members over 482 981 windows; the fine-tuning pass consumed them at a confidence threshold of 0.7 (315 087 positive (window, class) entries, 3.8 positives per soundscape window), as detailed in Section 8. INT8 and float16 export attempts and their failures are reported in Section 10; the shipped models are float32 ONNX graphs.

10. Inference Under the Runtime Budget

The CPU runtime cap on the hidden test set bounded the ensemble. We report the conversions that worked and the ones that did not, because the budget shaped which models could ship.

- A weighted seven-model soundscape-tuned ensemble exceeded the runtime cap and timed out on the hidden test set; the submitted system uses a three-member triplet instead.

- A ConvNeXt-Base soundscape-tuned model scored well locally but projected to roughly 114 minutes on the hidden test set, above the cap, and was shelved.
- Two attempts to make the heavier models fit failed: dynamic INT8 quantisation of the B3 ONNX graph ran about five times slower and produced near-zero agreement with the float model, an interaction with the depthwise convolutions; and float16 conversion failed with a type mismatch in the converter that `keep_io_types` did not resolve.

The practical consequence is that the submitted ensemble is built from the models that both helped on validation and fit the budget, which excluded our best locally scoring single model.

11. Component Ablation

Two readings isolate where the score comes from. On the local harness (Table 1), the soundscape-tuned triplet members saturate the macro-average, and the measurable gain from combining them with the public SED branch is in the worst-3 tail. On the leaderboard (Table 3), the soundscape-tuned triplet adds a small, repeatable lift over the public baseline: the baseline alone scored 0.927, and rank-blending the triplet onto it at $\alpha = 0.10$ reached 0.932.

The blend weight α has a clear optimum. Sweeping it for the B3 member alone gave 0.930 at $\alpha = 0.07$, 0.932 at $\alpha = 0.10$, 0.931 at $\alpha = 0.12$, and 0.923 at $\alpha = 0.20$: the soundscape-tuned branch is a useful correction at low weight and degrades the baseline when it is allowed to dominate. We read this as the baseline being the stronger absolute model on the hidden distribution, with the soundscape-tuned triplet supplying complementary signal that is only safe in small doses.

12. Failed Experiments

We document directions that did not improve the leaderboard, to inform future submissions.

Argmax one-hot pseudo-labels. An earlier pseudo-labelling pass used a single-class argmax target per window. Models trained on it plateaued at 0.928–0.930, below the 0.932 peak, regardless of the confidence threshold or the pseudo-label weight. The multi-hot pseudo-labels of Section 8 were the response to this plateau.

Distillation into a single student. Distilling a seven-model soundscape-tuned teacher into one B3 student scored 0.925, below the teacher and below the triplet. The runtime budget motivated this, but the student lost more than the budget saved.

Geometric-mean fusion. Replacing the rank-blend of the triplet with a geometric mean of probabilities scored 0.924, below the rank-blend at the same composition. The component models are not calibrated to a shared scale, which is the condition under which rank-blending is the safer fusion.

Higher triplet weight. Raising α above the 0.10–0.12 plateau reduced the score (Section 11); the soundscape-tuned branch does not improve the baseline when it carries more than a small share of the blend.

A multi-hot pseudo model held back. One soundscape-tuned model trained on multi-hot pseudo-labels tied the previous best on the local harness and was not submitted, to avoid spending a leaderboard slot on a configuration the harness predicted to be neutral.

13. Submitted Runs and Results

Table 3 lists representative submissions grouped by method, with both public- and private-leaderboard macro-AUC. We submitted 52 entries over the competition; the table reports one per method family. The best public-leaderboard score among all our submissions was 0.93215, placing 1968th of 4092 teams. The competition scores a small number of author-selected submissions on the private leaderboard; we selected the weighted soundscape-tuned triplet, which scored 0.91710 privately (2415th of 4092) and was the best of our selections. The row labelled *last-submitted* is the final entry we uploaded chronologically (a half-window test-time-augmentation variant); it was not the selected entry and is listed only for completeness, which is why it appears below the selected triplet despite the row order otherwise following date. The drop of roughly 450 places between the two boards is the subject of the next subsection.

Table 3
Representative leaderboard submissions, grouped by method, with public- and private-leaderboard macro-AUC. Rows are ordered roughly by date. The public and private columns do not rank the rows in the same order: the highest public score (the soundscape-tuned triplet) is not the highest private score.

Submission	Public LB	Private LB
Public baseline (Perch 0.7 + SED 0.3, rank-blend)	0.92687	0.91116
Selective per-class blend over the public baseline	0.92387	0.92262
+ B3 soundscape-tuned ($\alpha = 0.10$)	0.93192	0.91468
Equal-weight soundscape-tuned triplet ($\alpha = 0.10$)	0.93215	0.91664
Weighted soundscape-tuned triplet ($\alpha = 0.10$)	0.93209	0.91710
+ cross-year-pretrained B3 as 4th anchor (quad)	0.93148	0.91610
Retrained embedding-branch base (standalone)	0.92642	0.92039
Retrained embedding base + soundscape triplet	0.92680	0.91042
Geometric-mean fusion of the triplet	0.92382	0.90905
Last-submitted: triplet + half-window TTA	0.92812	0.91097

13.1. The public-to-private gap

The two leaderboards disagree on which submission was best. The soundscape-tuned triplet that topped our public scores (0.932) scored 0.917 on the private split. Two configurations with clearly lower public scores scored higher privately: a selective per-class blend over the public baseline (0.924 public, 0.923 private) and a standalone retrained embedding branch (0.926 public, 0.920 private). Our per-blend and per-threshold choices were made by maximising public-leaderboard macro-AUC, which is itself a form of fitting to the public split, and the gap between the two columns is the cost of that procedure. We did not have an independent held-out estimate of the private distribution: the local harness measures the soundscape-tuned and SED parts well but cannot hold out the embedding branch that carries most of the score, so it could not have predicted this inversion. A non-leaky cross-validated estimate of the full blend, including the embedding branch, is the change most likely to have closed this gap, and we flag it as the main methodological shortcoming of the entry.

13.2. What was validated non-leakily, and a protocol that would fix the rest

It is worth separating the choices that were made on a clean held-out estimate from those that were fit to the public leaderboard. Validated non-leakily on the harness (out-of-fold over the 66 soundscape files): the composition and rank-mean weights of the soundscape-tuned triplet, the decision to add the public SED branch, and the shape of the α operating point on the worst-3 tail. Selected on the public leaderboard, and therefore fit to the public split: the embedding-branch weight (0.7) within the baseline, the per-class selective blend, the inference threshold, and the final value of α. The public-to-private

inversion in Table 3 is confined to the second group, because the harness can rank everything in the first group but nothing in the second.

The leak is structural rather than incidental: in submission mode the Perch probe head (and its state-space correction) is fit on all 66 labelled soundscape files, so any harness reading that includes the embedding branch has already seen its own evaluation labels. A non-leaky estimate of the full blend would re-fit that branch out-of-fold. Concretely, under the same file-level 5-fold split: for each held-out fold, re-fit the probe head and the state-space correction on the soundscape windows of the other four folds only, predict the SED and triplet branches out-of-fold as already done, and score the complete blend (all branch weights, α, and the threshold) on the held-out fold. Averaging over folds gives a held-out macro-AUC of the entire system, including the dominant branch, on which the blend weights and threshold could be chosen instead of on the public leaderboard. We did not implement this in time; it is the single change we would prioritise for a future entry.

14. Limitations

The submitted system has four limitations. Most of the final score comes from a public baseline that we adopted unchanged, and our own contribution is a small, complementary correction. The development track that would have let our own models carry more of the score was not completed before the deadline, bounded by the training throughput of the two machines described in Section 8. The blend weights and the operating threshold were tuned against the public leaderboard, and the public-to-private inversion in Table 3 shows that this tuning did not transfer. The local harness is restricted to the 75 species with positive labels in the soundscape set, so it is blind to the other 159 classes and gives no clean held-out estimate of the embedding branch, which trains on those same files. And the inference-runtime budget excluded our strongest single local model and capped the ensemble size, so the entry does not reflect the best model we trained, only the best one that fit.

15. Code and Data Availability

The released BirdCLEF+ 2026 data, sample submission, and leaderboard are available through the official Kaggle competition page [1]. The components of the submitted system have the following provenance. The Perch-embedding branch and the five-fold SED ensemble are publicly shared community baselines used unchanged; the Perch model itself is openly available [7]. The soundscape-tuned triplet (EfficientNet-B3, a pseudo-refined B3, and ConvNeXt-Small) was trained by the author on the released training and soundscape audio, exported to ONNX for CPU inference, and rank-blended onto the baseline at $\alpha = 0.10$. The cross-year-pretrained B3 of Section 8 was trained on a corpus merged from the 2026 and 2025 BirdCLEF releases.

Pipeline source code, the per-model configurations, and the blend specification are released at https://github.com/gcol33/bird-clef-2026 under an MIT licence; the trained ONNX weights for the soundscape-tuned triplet and the cross-year-pretrained B3 are released on the Hugging Face Hub at https://huggingface.co/gcol33/bird-clef-2026 under CC-BY-4.0. Audio data is not redistributed and must be obtained from the Kaggle competition page. The publicly shared Perch and SED baselines remain under their original authors' terms and are referenced rather than redistributed.

16. Conclusion

Our BirdCLEF+ 2026 entry blends a publicly shared acoustic baseline with a weighted triplet of soundscape-fine-tuned convolutional networks, rank-blended at $\alpha = 0.10$. It scored 0.93215 macro-AUC on the public leaderboard (1968th of 4092 teams) and 0.91710 on the private leaderboard (2415th of 4092). The component ablation locates the contributions: the public baseline carries the bulk of the score, the soundscape-tuned triplet adds a small repeatable lift that is safe only at low blend weight, and the gain from the triplet is concentrated in the worst-performing classes. The most useful piece

of infrastructure was the local soundscape harness, which moved hypothesis testing off the daily leaderboard slot.

Two lessons carry forward. The development track confirms that the route to the higher leaderboard range in this task is a deeper data pipeline (cross-year pretraining, frame-level multi-hot pseudo-labels, and a multi-backbone zoo) rather than a better blend of fixed components, and that this pipeline has to be started early because it does not fit a final week. And the public-to-private inversion is a reminder that an operating point tuned on the public split carries its own error; the fix we flag in Section 13 is a non-leaky cross-validated estimate that includes the dominant branch.

Acknowledgments

We thank the BirdCLEF and LifeCLEF organisers for releasing the data and running the leaderboard, the Cornell Lab of Ornithology and the wider community for the Perch and BirdNET models, and the authors of the public baseline notebooks whose work the submitted system builds on. The competition dataset was developed with support from the Bezos Earth Fund AI for Climate and Nature Grand Challenge and annotated by domain experts across the contributing institutions.

Declaration on Generative AI

During the development of the system, the author used a local REAP-48B model for coding assistance (typing speedup and routine code transformations); REAP-48B is a 48-billion-parameter sparse mixture-of-experts checkpoint derived from Qwen3-Next-80B [16] via the REAP expert-pruning method [15], served on an Apple M4 Pro. No proprietary or remote LLM was used. The machine-learning design and the prose of this manuscript are the author's own work; the author takes full responsibility for the publication's content.

References

[1] S. Kahl, T. Denton, L. Sugai, L. Piatti, R. Holbrook, H. Klinck, and A. Oldacre. BirdCLEF+ 2026. Kaggle, 2026. https://kaggle.com/competitions/birdclef-2026

[2] ImageCLEF / LifeCLEF. BirdCLEF+ 2026. https://www.imageclef.org/BirdCLEF2026

[3] L. Picek, L. Adam, S. Kahl, R. Bossy, L. Chrobak, H. Goëau, K. Papafitsoros, H. Klinck, W.-P. Vellinga, R. Planqué, T. Denton, K. Barnard, C. Nédellec, L. Deléger, M. Courtin, G. Martellucci, I. Moummad, F. Vinatier, P. Bonnet, and A. Joly. Overview of LifeCLEF 2026: AI challenges for biodiversity understanding and ecosystem management. In *International Conference of the Cross-Language Evaluation Forum for European Languages (CLEF)*. Springer, 2026.

[4] Stefan Kahl, Tom Denton, Larissa Sugai, Liliana Piatti, Wener Hugo Arruda Moreno, Mariana Motti Barbosa, Maximilian Eibl, Carolline Zatta Fieker, Carolina Martins Garcia, Daiene Louveira Hokama Sousa, João Emílio de Almeida Júnior, Ryan Christopher Kridler, Mario Lasseck, Alyson Vieira de Melo, Henning Müller, Matheus de Oliveira Neves, Matheus Gonçalves dos Reis, Lucas Korzune Sampaio Teles, Karl-L. Schuchmann, José Luiz Massao Moreira Sugai, Kirk Thiago Pedroso Azevedo, Priscila do Nascimento Lopes, Marinez Isaac Marques, Holger Klinck, Hervé Glotin, Hervé Goëau, Willem-Pier Vellinga, Robert Planqué, and Alexis Joly. Overview of BirdCLEF+ 2026: Acoustic species identification in the Pantanal, South America. In *Working Notes of CLEF 2026 – Conference and Labs of the Evaluation Forum*, 2026.

[5] J. S. Cañas, S. Kahl, T. Denton, M. P. Toro-Gómez, S. Rodriguez-Buritica, J. L. Benavides-Lopez, J. S. Ulloa, P. Caycedo-Rosales, H. Klinck, H. Goëau, W.-P. Vellinga, R. Planqué, and A. Joly. Overview of BirdCLEF+ 2025: Multi-taxonomic sound identification in the Middle Magdalena, Colombia. In *Working Notes of CLEF 2025 – Conference and Labs of the Evaluation Forum*, pages 2909–2919, 2025.

[6] S. Kahl, T. Denton, H. Klinck, V. Ramesh, V. Joshi, M. Srivathsa, A. Anand, C. Arvind, H. CP, S. Sawant, V. V. Robin, H. Glotin, H. Goëau, W.-P. Vellinga, R. Planqué, and A. Joly. Overview of BirdCLEF 2024: Acoustic identification of under-studied bird species in the Western Ghats. In *Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum*, pages 1948–1957, 2024.

[7] J. Hamer, E. Triantafillou, B. van Merriënboer, S. Kahl, H. Klinck, T. Denton, and V. Dumoulin. Feature embeddings from large-scale acoustic bird classifiers. *arXiv preprint arXiv:2307.06292*, 2023. https://doi.org/10.48550/arXiv.2307.06292

[8] S. Kahl, C. M. Wood, M. Eibl, and H. Klinck. BirdNET: A deep learning solution for avian diversity monitoring. *Ecological Informatics*, 61:101236, 2021. https://doi.org/10.1016/j.ecoinf.2021.101236

[9] M. Tan and Q. V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proc. 36th International Conference on Machine Learning*, pages 6105–6114, 2019.

[10] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie. A ConvNet for the 2020s. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022. https://doi.org/10.1109/CVPR52688.2022.01167

[11] T. Ridnik, E. Ben-Baruch, N. Zamir, A. Noy, I. Friedman, M. Protter, and L. Zelnik-Manor. Asymmetric loss for multi-label classification. In *Proc. IEEE/CVF International Conference on Computer Vision*, pages 82–91, 2021. https://doi.org/10.1109/ICCV48922.2021.00015

[12] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le. SpecAugment: A simple data augmentation method for automatic speech recognition. In *Proc. Interspeech*, pages 2613–2617, 2019. https://doi.org/10.21437/Interspeech.2019-2680

[13] R. Wightman. PyTorch image models. https://github.com/huggingface/pytorch-image-models, 2019.

[14] scikit-learn developers. `sklearn.metrics.roc_auc_score`. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html

[15] M. Lasby, I. Lazarevich, N. Sinnadurai, S. Lie, Y. Ioannou, and V. Thangarasa. REAP the experts: Why pruning prevails for one-shot MoE compression. *arXiv preprint arXiv:2510.13999*, 2025. https://doi.org/10.48550/arXiv.2510.13999

[16] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. https://doi.org/10.48550/arXiv.2505.09388