Mixup Consistency Regularization for Noise-Robust Bioacoustic Classification for BirdCLEF 2026

BirdCLEF+ 2026 at CLEF 2026

Antoine Masquelier

Abstract

This working note describes our approach for the BirdCLEF+ 2026 competition, where we achieved 12th place as team *Bobbing Redstart* out of 4,094 teams in the challenge of identifying 234 wildlife species from soundscapes in Brazil's Pantanal wetlands. Automating species identification from these complex field recordings presents significant challenges, including heavily imbalanced long-tailed distributions, incomplete annotations, and a substantial domain gap between clean, focal training recordings and noisy, multi-species test soundscapes. To address both the noisy nature of the labels and this domain gap, we propose a framework centered on a Mixup consistency regularization mechanism, paired with a class-wise asymmetric loss that accounts for species-specific label noise. Our approach is grounded in the principle that audio mixing is label-preserving; any species present in an individual source recording is assumed to remain present in the resulting mixture. By enforcing that the model's predictions on mixed individual inputs remain consistent with the combined predictions of the corresponding individual inputs, our framework aims to mitigate the impact of missing labels and improve robust generalization to complex, real-world soundscapes. We find that these two components contribute little in isolation but yield a pronounced and robust improvement when combined. The code used for this work can be found at https://github.com/AMasquelier/BirdCLEF2026.

Keywords

Bioacoustics, Noisy Labels, Consistency Regularization, Semi-Supervised Learning, BirdCLEF

1. Introduction

Passive acoustic monitoring (PAM) is an increasingly important tool for biodiversity assessment, enabling continuous, non-invasive observation of wildlife across large spatial and temporal scales. The BirdCLEF competition[1, 2, 3, 4, 5, 6, 7, 8] has been held annually since 2014, and hosted on Kaggle since 2020. This year's edition[9, 10] focuses on identifying 234 wildlife species including birds, amphibians, mammals, insects, and reptiles in the Pantanal. The Pantanal is a wetland spanning more than 150,000 km^2 across multiple countries, where cycles of rainy and dry seasons completely reshape the landscape.

However, the nature of the available annotated data raises different challenges:

- Recordings typically contain vocalizations from multiple co-occurring species, but annotations are often incomplete: only the focal species and occasionally some secondary species are labeled, while other audible species in the background remain unannotated. This results in systematic false negatives, where the absence of a label does not reliably indicate the absence of a species.

- Training recordings vary in length (from a few seconds to several minutes), but models operate on short fixed-duration crops (5 seconds in our case). When using a recording-level label for a randomly sampled crop of the recording, the target species may not be vocalizing within that particular temporal window, producing ground-truth false positives.

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany
antoine.masq@gmail.com (A. Masquelier)
https://github.com/AMasquelier (A. Masquelier)
0009-0004-4976-1254 (A. Masquelier)

- The number of available recordings for each species can vary greatly and thus results in a highly imbalanced long-tailed distribution.

- While the annotated training data are mostly focal recordings with a clear signal for the target species from various locations, the test data are noisy soundscapes from the Pantanal with multiple species vocalizing simultaneously and environmental noises like wind or rain. This creates a domain gap between training and test conditions, as models trained on clean focal recordings must generalize to noisy, multi-species soundscapes where species co-occurrence patterns and prevalence rates differ substantially.

In this work, we address the noisy nature of the labels and the domain gap between training and test data through a consistency regularization framework grounded in the assumption that audio mixing is label-preserving. Specifically, any species present in an individual source recording is assumed to remain present in the resulting mixture. Consequently, predictions on mixed individual inputs should be consistent with the combined predictions on the corresponding individual inputs. Since such mixtures emulate the multi-species, noisier character of the target soundscapes, this consistency objective also encourages robustness to the conditions encountered at test time.

Additionally, we extend the asymmetric loss (ASL)[11] with class-wise noise assumptions, adjusting its per-class behavior to reflect that label noise, particularly missing positives, varies across species.

2. Related work

Over the past editions of BirdCLEF, the dominant and consistently successful paradigm has been to treat bioacoustic classification as a computer vision problem, applying convolutional or transformer-based image models to mel spectrogram representations of the audio signal. This approach underlies most state-of-the-art systems, including BirdNet [12] and Perch [13, 14]. Our work follows this well-established paradigm and focuses instead on the training objective and regularization strategy.

A recurring challenge across the different editions of the BirdCLEF competitions has been the domain gap between the clean focal training recordings and the noisy soundscapes encountered at test time. Another difficulty has been the noisy nature of the training data, where recordings were annotated for target species but background species remained unannotated, compounded by the fact that those models often work on limited recording duration (e.g. 5 seconds windows) which might create false positives when sampling from larger training recordings.

To address such generalization and noisy label issues, data augmentation techniques like Mixup [15] have been widely adopted in environmental audio classification and solutions from previous BirdCLEF editions. Unlike image mixtures, mixing audio signals linearly is a physically sound operation that naturally simulates multi-species soundscapes. While standard Mixup improves model robustness and generalization, it fundamentally relies on the provided ground-truth labels, which in this setting are often weakly annotated.

Consistency regularization has served as a fundamental framework in semi-supervised learning [16, 17]. This technique typically encourages a model to produce invariant predictions across differently perturbed versions of the same input. Recent methodologies have extended this concept by combining it with Mixup[18, 19], effectively constraining predictions on interpolated data points without relying strictly on ground-truth labels. Building on this literature, our work introduces a tailored Mixup consistency regularization mechanism that is used in both supervised and semi-supervised settings.

3. Task and evaluation

The competition is a multi-label classification problem over 234 wildlife species spanning five taxonomic classes (Aves, Amphibia, Mammalia, Insecta, and Reptilia). The hidden test set consists of roughly 600 one-minute long soundscapes, with a 66 and 34% split between the private and public sets. This year, each of those soundscapes contains metadata about the site, time of the day and date of the recording. At inference, each one-minute test soundscape is divided into twelve consecutive 5-second segments, and for every segment a model must output a presence probability for each of the 234 classes. Submissions are evaluated with a macro-averaged ROC-AUC that skips classes that have no true positive labels[1]. An additional constraint is that the inference on this test set should run on CPU within a 90-minute time limit in a Kaggle notebook.

4. Data

4.1. Overview

4.1.1. Focal audio recordings

The training data is composed of 35,549 audio recordings with one primary label and, occasionally, secondary labels. The distribution across species is strongly imbalanced and long-tail-shaped. The recordings come from two sources: Xeno-Canto [20] (65% of the recordings) and iNaturalist [21] (35% of the recordings). It should be noted that only the data coming from Xeno-Canto has secondary labels. All the recordings were resampled at 32 kHz and stored in the ogg format. The duration of those vary between 2.4 seconds and 1.9 hours, with an average duration of 35 seconds.

4.1.2. Soundscapes

The test data is composed of soundscapes recorded at various sites and times across the Pantanal. A particularity of this year's edition is that it includes metadata associated with the sites, times of day and recording dates of the recordings. Although the test data is not available, there are 10,658 unlabeled soundscapes and 66 labeled soundscapes that are available in the training data. The soundscapes from both training and test data are all 60 seconds long. A further specificity of this year's dataset is that certain species are absent from the focal recordings in the training set and are instead represented exclusively within the annotated soundscape data. Specifically for the class Insecta, the soundscapes are annotated only with insect sonotypes rather than with precise species-level identifications. The fact that this type of annotation is unique to the soundscapes introduces an additional challenge in terms of domain shift. When comparing the class distributions between the focal and labeled soundscapes (see Table 1), a clear shift in the distribution is apparent. Whereas the vast majority of focal recordings are annotated with Aves labels, the soundscapes predominantly contain Amphibia labels, and the degree of class imbalance is less pronounced in the soundscapes than in the focal recordings.

Table 1
Classes distribution across labeled training data

Class name	Focal recordings	Soundscapes
Aves	97.890 %	20.270 %
Amphibia	1.269 %	59.460 %
Insecta	0.560 %	15.405 %
Mammalia	0.279 %	2.703 %
Reptilia	0.003 %	2.162 %

[1] https://www.kaggle.com/code/metric/birdclef-roc-auc

4.2. Additional data

As noted by Patrick Robitaille and Konstantin Dimitriev on the competition forum[2], one species was under-sampled because the *Antiurus maculicaudus* taxon has an inactive taxon synonym (*Hydropsalis maculicaudus*) that was still used on Xeno-Canto. We then decided to verify this for other under-sampled species and found that it was also the case for *Rufirallus schomburgkii* that have a synonymous taxon used on Xeno-Canto (*Micropygia schomburgkii*). Additionally, we found 6 other recordings for *Canis familiaris*. Overall, we retrieved from Xeno-Canto 91 recordings of *Rufirallus schomburgkii*, 72 of *Antiurus maculicaudus*, and 6 of *Canis familiaris*.

4.3. Xeno-Canto dataset for pretraining

We collected data from Xeno-Canto for a total of 296,052 recordings. Those recordings span 10,556 species and 2,717 genera for birds, grasshoppers, frogs, land mammals and bats. The lengths of the recordings go from recordings less than a second to a maximum length of 4 minutes. Only the genera with more than 32 examples are kept.

4.4. Preprocessing

4.4.1. Corrupted files

During the data exploration, we noticed that some recordings were shorter than 1 second and their content seemed corrupted. We re-downloaded those by using the provided urls, and it resulted in healthy recordings with varying length. In total, 369 files were found to be corrupted for a total of around 2 hours and 50 minutes of data.

4.4.2. Manual cleaning

We also noted that some minority classes had long recordings with unlabeled species calling in the background while the target species were doing short punctual calls. The best example of this is for the *Bos taurus*, where there are only 10 available recordings including 6 from a single recordist, all with the same unlabeled background species. This might result in the model confusing the background species for the target species. We decided to manually remove those time windows for the following species : *Bos taurus*, *Canis familiaris*, and *Mico melanurus*.

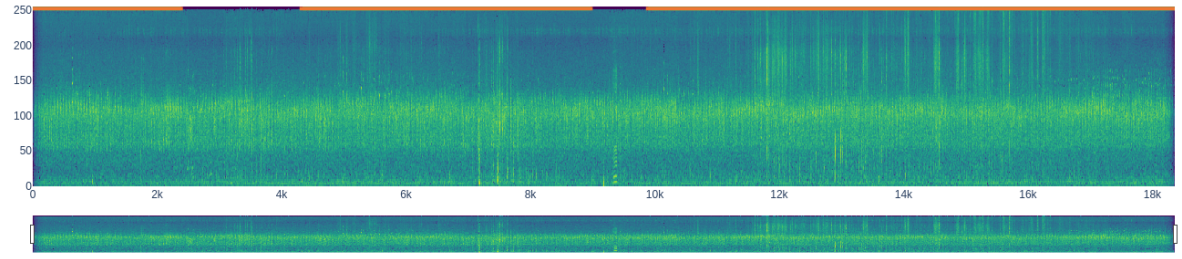

Figure 1: Mel spectrogram of a *Bos taurus* recording (XC1067100). The orange bars on top show the time windows during which the target species is not calling but unlabeled background species does.

4.4.3. Duplicated files

We also computed hash values for the raw signals with PyTorch's *hash_tensor*[3] function. The results showed that 86 unique recordings were duplicated up to 2 times, generating 90 duplicated recordings in

[2]https://www.kaggle.com/competitions/birdclef-2026/discussion/681005
[3]https://docs.pytorch.org/docs/2.12/generated/torch.hash_tensor.html

total. The majority of the duplicated files were coming from the iNat collection, and only one of them from Xeno-Canto. The duplicated files were then excluded from the dataset to avoid leaking between the training and test sets.

4.4.4. Silent soundscapes

Focal recordings always contain target species by definition but soundscapes do not necessarily contain calls from target species, some of them only contain background noises like wind, rain, or just silence. Training only on examples with positive labels would likely bias the model toward positive predictions. To identify likely-silent soundscapes, we used early versions of our models to predict on the unlabeled soundscapes and ranked them by their maximum predicted probability across all species and all twelve 5-second windows. We then reviewed candidates manually in ascending order of this maximum, starting from the most confidently empty recordings. Rather than selecting a principled decision boundary, we treated the maximum probability purely as a review-ordering heuristic and stopped at approximately 0.3, beyond which recordings increasingly contained audible calls and manual verification became less productive. Every retained soundscape was manually confirmed to be empty before being added to the training data with no labels. A model was trained with these and was used to flag potential other silent soundscapes that were manually reviewed again. In total, 32 soundscapes were flagged as silent.

4.5. Sampling

4.5.1. File sampling

To address the highly imbalanced data distribution, we decided to sample as described in equation 1 which is similar to Sydorskyi & Gonçalves[22] working note for last year's edition.

$$w_i = \left(\frac{c_i}{\sum_j c_j} \right)^{-\beta} \tag{1}$$

where c_i is the count of species i.

The weights are only computed based on the training focal recordings. Subsequently, the annotated and silent soundscapes, and additional recordings described in the previous sections are assigned the average weight value. A value of 0.8 was used for β to train all our models.

4.5.2. Signal sampling

We decided to keep the same granularity as the test data, i.e. 5 seconds windows. In the case where recordings are longer than 5 seconds, we randomly select a 5 seconds window within the recording following a uniform distribution. In the opposite case where a recording is shorter than 5 seconds, we randomly place it within an empty 5 seconds recording with a uniform distribution.

5. Validation strategy

One of the most challenging aspects of the last BirdCLEF competitions has been to find a reliable validation strategy because of the domain shift between the training and test data. However, in this year's edition, a small amount of annotated soundscapes has been made available in the training data. These annotated soundscapes are also important for training as some species only appears in those annotated soundscapes. The main challenge in this edition is then to balance the use of those annotated soundscapes between the training and validation set.

For the focal recordings, our validation strategy was a stratified 5-folds split based on the primary labels. The additional data, silent soundscapes are not split and are all used for training.

The labeled soundscapes were split in a specific way to ensure all species are present in each fold. More

specifically, the recordings from sites 3 and 18 were kept as validation data only, which means that those are never used for training except when explicitly training on the whole dataset. Recordings from site 22 are randomly split with 70% allocated for training the rest being used for validation, and recordings from site 8 are split with 1 random recording allocated for validation and the rest allocated for training.

6. Model

The model is composed of an EfficientNetV2[23] backbone and a simple classification head composed of 2 dense layers with a hidden dimension of 512, followed by a Leaky ReLU[24] activation function and a Dropout of 0.3. More specifically, the backbones used in this work are the tf_efficientnetv2_b0, tf_efficientnetv2_b3 and tf_efficientnetv2_s from the timm[25] Python library.

7. Loss function

The loss used in this work is a modified version of Asymmetric Loss (*ASL*)[11], which we refer to as Class-wise ASL (*CASL*).

The primary motivation behind the original *ASL* formulation (equation 2 and 3) is addressing the severe positive-negative imbalance inherent to multi-label classification: each sample typically contains only a few positive labels alongside numerous negative ones. Consequently, the accumulated gradients from easy negative samples tend to dominate training and suppress the contribution of positive instances. *ASL* mitigates this by applying asymmetric focusing, which down-weights easy negatives more aggressively than positives.

$$L = -y\, L_+ - (1-y)\, L_- \tag{2}$$

$$\text{ASL} = \begin{cases} L_+ = (1-p)^{\gamma_+} \log(p) \\ L_- = (p_m)^{\gamma_-} \log(1-p_m) \end{cases} \tag{3}$$

where $p_m = \max(p - m,\, 0)$

In this work, we briefly explore how a class-wise tuning of the γ_+ and γ_- parameters can help to softly mitigate the label noise. During our data exploration we noticed that certain background species, especially insects, were frequently left unannotated by recordists while the species were present. Lacking precise quantitative measurements of these missing-label rates, we relied on qualitative domain assumptions to manually assign γ_+ and γ_- values. For classes assumed to have a high likelihood of unannotated background presence (false negatives), a high γ_- was assigned to softly attenuate the destructive gradients that occur when the network correctly detects an unannotated acoustic signal. Similarly, classes assumed to have a high likelihood of cut-off calls (false positives) were assigned a higher γ_+. Conversely, for classes assumed to have reliable, clean negative annotations, a lower γ_- was preserved to prioritize learning from trustworthy negative boundaries. To streamline the hyperparameter space for this brief exploration, the probability margin p_m from the original formulation was omitted by setting $p_m = 0$.

While not explicitly discarding mislabeled samples, which the original formulation achieves via the probability margin p_m, this asymmetric focusing attenuated the massive volume of negative gradients. This overall reduction prevented the prevalent unannotated background signals from overpowering the scarce, reliable positive annotations. The complete settings for the γ values can be found in Table 2.

Table 2
CASL gamma parameters

	Aves	Amphibia	Insecta	Mammalia	Reptilia
γ_+	0.5	0.0	0.0	0.5	0.0
γ_-	0.5	0.5	2.0	0.1	0.5

Although we did not carry out a formal tuning of these per-class parameters, we compared this qualitative configuration against two different ASL configurations in Table 12. The ASL settings yields only a marginal improvement over BCE, whereas our class-wise configuration produces a clear gain in soundscape ROC-AUC as well as on both leaderboard scores, which suggests that the specific per-class values, despite being manually assigned, capture a meaningful signal beyond what a single global asymmetric-focusing setting can provide.

Additionally, we ran a controlled single-class ablation, reported in Table 13, in which the negative focusing parameter γ_- is set to 2 for a single class at a time while all remaining γ_+ and γ_- values are held at 0. We did not perform any formal search over these values; this experiment instead isolates the effect of attenuating the negative gradients of each class independently. The largest gain comes from Insecta, the class we had a priori identified as the most frequently left unannotated in the background. This retrospectively supports the intuition behind our manual assignment, since the class for which suppressing negative gradients helps most is precisely the one where unlabeled background presence is expected to be most prevalent.

In addition to that custom loss function, we down-weight (by a factor of 0.1) the loss values for the sonotype labels on the focal recordings, since those labels are specific to the soundscapes.

8. Augmentation

8.1. Stretch

The *Stretch* augmentation takes the spectrogram as input and simulates time-stretching or pitch-shifting effects by randomly cropping a portion of the spectrogram along the selected axes (e.g., the time axis, or both time and frequency axes) and resizing the cropped image back to its original dimensions. This artificially introduces variations in call durations and frequency footprints with minimal computational overhead.

8.2. Masking

Our *Masking* augmentation builds upon SpecAugment [26]. When applying classical SpecAugment, which uses continuous bands spanning the entire time or frequency axis, we intentionally configure the bands to be very thin. This careful tuning distorts the spectrogram features just enough for effective regularization, while avoiding the risk of completely masking brief or narrow-band bird calls.

Furthermore, our implementation extends this standard approach by allowing the random dropout of localized rectangular patches instead of full-axis bands. Depending on the augmentation intensity, we apply multiple masks (typically between 8 and 16) with dimensions ranging from 0 to 3 pixels for bands and 0 to 16 pixels for rectangular patches. This combined approach of thin bands and localized patches encourages the network to learn robust, distributed representations rather than overfitting to specific time-frequency structures.

8.3. Mixup

The Mixup [15] augmentation has shown great potential for audio classification [27] and acts as the core regularization mechanism of this work. Traditionally, Mixup is defined as a random linear interpolation of both the inputs and the labels of different data points. However, to better reflect the physical reality of overlapping audio environments as suggested by van Merriënboer et al. [14], we apply linear interpolation to the input signals but use a maximum operator for the labels:

$$\tilde{x}_i = \lambda\, x_i + (1-\lambda)\, x_j \tag{4}$$

$$\tilde{y}_i = \max(y_i, y_j) \tag{5}$$

$$\lambda \sim Beta(\alpha, \alpha) \tag{6}$$

It should be noted that this label-preserving assumption is an idealization: acoustic masking, where a louder or spectrally overlapping source obscures a quieter one, can in practice reduce the perceptual salience of a species in the mixture, even though it remains physically present in the signal.

9. Training pipelines

9.1. Supervised learning

Our supervised learning baseline, illustrated in figure 2, is built around a Mixup consistency regularization mechanism inspired by MixMatch [18]. The inputs x and y denote, respectively, a batch of 5-second signals and their multi-label targets.
We denote by Model* the full forward operator, which transforms the raw signal into a log-mel spectrogram, applies the stochastic augmentations described in section 8, and produces class logits through the model described in section 6. We write $\hat{y}$ for the prediction on an individual input.
Within each batch, we form mixed inputs $\tilde{x}$ and mixed targets $\tilde{y}$ following our label-preserving Mixup formulation (equation 4), where signals are linearly interpolated but labels are combined through a maximum operator.
The mixed predictions of the individual inputs $\hat{y}$ are defined as $\tilde{\hat{y}} = max(\hat{y}_i, \hat{y}_j)$, and the predictions over the mixed input are defined as $\hat{\tilde{y}} = Model^*(\tilde{x})$.

The total objective combines three terms. The first, L_S, is the standard supervised term that compares the prediction on individual inputs with their corresponding ground-truth labels. The second, L_M, is the supervised term on mixed inputs, where the targets are the max-combined labels. The third, L_C, is the consistency term, which enforces that the predictions on mixed inputs $\hat{\tilde{y}}$ agree with the mixed predictions obtained from the corresponding individual inputs $\tilde{\hat{y}}$ that are treated as fixed targets through a stop-gradient operation.

$$L_S = \text{CASL}(y, \hat{y}) \tag{7}$$

$$L_M = \text{CASL}(\tilde{y}, \hat{\tilde{y}}) \tag{8}$$

$$L_C = \text{CASL}(\tilde{\hat{y}}, \hat{\tilde{y}}) \tag{9}$$

$$L = L_S + L_M + (0.5 + \eta) L_C \tag{10}$$

where $\eta = \frac{\text{epoch}}{\#\text{epochs}}$.

The consistency term L_C leverages the additive physical property of sound to generate a self-supervised training signal. By treating the aggregated predictions from individual recordings as fixed targets for a mixed input, the model enforces a strict internal consensus. This serves a dual purpose:

first, it acts as a stabilizing anchor that resists the disruptive gradients and memorization caused by incomplete or noisy annotations. Second, because mixing individual focal recordings emulates soundscapes, this consistency actively bridges the critical domain gap between our training focal recordings and the complex, multi-species soundscapes encountered during testing.

However, such a consistency regularization mechanism introduces a risk of confirmation bias. To mitigate this risk, we maintain a strong supervised signal throughout training. In addition, applying a ramped schedule over epochs enables the model to first develop robust representations before relying too much on its own predictions.

Also, it should be noted that the consistency is not only on Mixup but also on different augmentations as the forward passes include the augmentation module.

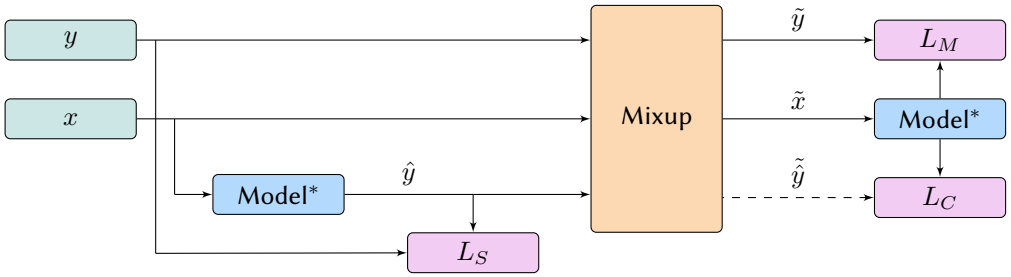

Figure 2: Overview of the supervised pipeline

9.2. Semi-supervised learning

Our semi-supervised training pipeline, illustrated in figure 3 extends the supervised baseline described in section 9.1 by incorporating the vast amount of unlabeled soundscapes described in section 4.1.2 into the same Mixup consistency framework.

Since the unlabeled soundscapes carry no annotations, we generate pseudo-labels using an ensemble of K teacher models obtained from the supervised-training round. Rather than precomputing a fixed pseudo-labeled set over the full unlabeled pool as in iterated self-training[28], we produce these targets online. For each unlabeled batch x_u, the pseudo-labels are computed as the average of the sharpened teacher predictions:

$$\hat{y}_u = \frac{1}{K} \sum_{k=1}^{K} \sigma \left(\frac{\text{Teacher}_k^*(x_u)}{T} \right) \tag{11}$$

where σ denotes the sigmoid function, K is the number of folds, and T is a temperature controlling the sharpness of the pseudo-labels. In this work, T was set to 0.5. These pseudo-labels are treated as fixed targets through a stop-gradient operation. This aggregation of the teachers predictions is represented by the $^+$ symbol in figure 3.

At each step, we draw a labeled batch (x_l, y_l) and an unlabeled batch x_u of the same size, and concatenate them into a single batch $x = [x_l, x_u]$ with combined targets $y = [y_l, \hat{y}_u]$. From this point, the pipeline follows the exact same structure as the supervised baseline: the same supervised term L_S, the mixed term L_M, and the consistency term L_C are computed over the concatenated batch following equations defined in section 9.1. The labeled portion thus retains its ground-truth supervision while the unlabeled portion is supervised by the teachers pseudo-labels, and the Mixup consistency mechanism

operates jointly over both, mixing labeled and unlabeled examples together.

This design lets the consistency objective act on genuine soundscape signals rather than only on synthetic mixtures of focal recordings, further narrowing the domain gap. As in the supervised setting, maintaining a strong supervised signal and ramping the consistency weight over epochs mitigates the confirmation bias inherent to training on a model's own predictions.

The teacher models consist of the five baseline models, each trained on a different fold.
It should also be noted that the student model is initialized using the weights of the teacher model from the corresponding fold.

As the model was initialized with the baseline weights, the total loss function uses a constant consistency term L_C:

$$L = L_S + L_M + L_C \tag{12}$$

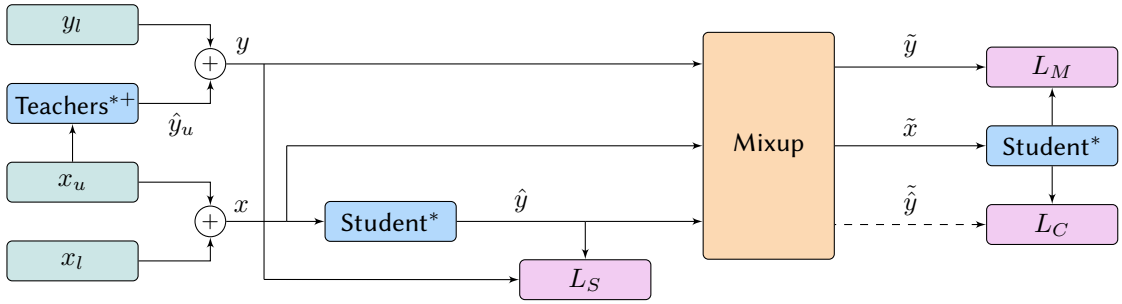

Figure 3: Overview of the semi-supervised pipeline

10. Pre-training

We trained models to predict the genus (2717 labels) on the data described in section 4.3 with the supervised training pipeline described in section 9.1. For each EfficientNetV2 variant employed in this study (tf_efficientnetv2_b0, tf_efficientnetv2_b3, and tf_efficientnetv2_s), a corresponding pre-trained model was trained. Nonetheless, the training configurations differed slightly, as summarized in Table 3. The reason behind those setup variations is to have diversity for the late stage ensembling.

Table 3
Pre-trained models

Backbone	L_C schedule	n_mel	f_min
tf_efficientnetv2_b0	L_C	224	20
tf_efficientnetv2_b3	L_C	224	20
tf_efficientnetv2_s	$(0.5 + \eta)L_C$	160	50

11. Baselines

To establish a strong foundation for the subsequent semi-supervised stage and to provide diverse teachers for pseudo-labeling, we trained several five-fold baseline models using the supervised pipeline described in section 9.1. The configurations, summarized in Table 4, vary the backbone, batch size, mel spectrogram parameters, augmentation intensity, and Mixup α. This diversity is deliberate: rather than tuning a single optimal setup, we sought complementary models whose disagreements would later

benefit both the ensemble and the teacher pool used for pseudo-labeling. All baselines were trained for 20 epochs with the optimizer and scheduler described in appendix A, and the scores reported correspond to those obtained by the ensemble of models trained on each of the five folds. It should also be noted that all the public and private leaderboard scores include the post-processing steps listed in section 13 excepted the priors. The listed augmentation regimes reflect the number of bands or squares that are applied in the Masking augmentation, as well as the scaling factor and axes employed in the Stretch augmentation (see section 8).

Across configurations, the tf_efficientnetv2_s baseline achieved the best private score (0.951), while the tf_efficientnetv2_b0 model with light augmentation and $\alpha = 2.0$ obtained the best public score (0.938). The gap between the public and private leaderboards, as well as between backbones, remained moderate, which suggests that no single configuration dominates and confirms the value of retaining several of them for ensembling.

Table 4
Baselines

ID	Backbone	Batch size	n_mel	f_min	Augmentation	Mixup α	Public	Private
1	tf_efficientnetv2_b0	32	224	50	light	2.0	**0.938**	0.940
2	tf_efficientnetv2_b0	32	224	50	strong	0.5	0.929	0.931
3	tf_efficientnetv2_b3	20	224	50	light	2.0	0.932	0.939
4	tf_efficientnetv2_s	32	160	50	light	1.0	0.936	**0.951**

To isolate the contributions of our two main methodological components, the class-wise asymmetric loss (CASL) and the Mixup consistency regularization term L_C, we conducted a series of ablation experiments reported in Table 5 where the Baseline 1 from Table 4 serves as a reference. We compare CASL against a standard BCE baseline, each with and without the consistency term, and report the mean and standard deviation of the local validation soundscapes ROC-AUC across the five folds alongside the public and private leaderboard scores.

When applied in isolation, each component yields only a modest local improvement, and the corresponding leaderboard movements are small and inconsistent: switching BCE to CASL without L_C leaves the soundscape ROC-AUC essentially unchanged, and adding L_C on top of BCE even slightly degrades it. In contrast, combining CASL with the consistency term produces a pronounced and consistent gain, raising the soundscape ROC-AUC and improving both the public and private scores, while also reducing the variance across folds. This synergy is the central empirical observation motivating our approach: the consistency mechanism is most effective when paired with a loss that already accounts for the asymmetric, class-dependent label noise. We attribute this to the fact that L_C amplifies the training signal derived from the model's own predictions, which is only beneficial when those predictions are not already corrupted by the destructive gradients that CASL is designed to attenuate.

To verify that these findings are not an artifact of a particular data ordering, we repeated the ablation with an alternative random seed affecting only data sampling and augmentation. The results, reported in Table 11 in the appendix, are qualitatively consistent: the combination of CASL and L_C again yields the best soundscape ROC-AUC and the best leaderboard scores.

To assess the contribution of the pre-training stage, we compared the five-fold baseline (Baseline 1 from Table 4) trained from ImageNet weights against the same configuration initialized from the genus pre-trained model, keeping every other setting fixed. As reported in Table 6, the pre-training greatly improves both leaderboard scores, while also tightening the variance across folds. We attribute this gain to the warm start providing a domain-adapted feature extractor. Having already learned to

Table 5
5-folds baseline loss function evaluation

Loss	L_C	Soundscape ROC-AUC	Public	Private
BCE	No	0.900 ± 0.041	0.926	0.931
BCE	Yes	0.894 ± 0.033	0.924	0.932
CASL	No	0.898 ± 0.036	0.925	0.931
CASL	Yes	**0.918 ± 0.024**	**0.938**	**0.940**

discriminate bioacoustic features, the model converges to a stronger soundscape representation than one bootstrapped from natural images.

Table 6
Evaluation of the effect of pre-training

Pre-training	Soundscape ROC-AUC	Public	Private
No	0.901 ± 0.026	0.928	0.927
Yes	**0.918 ± 0.024**	**0.938**	**0.940**

12. Pseudo-labeling

Building on the baseline teachers from section 11, we trained a collection of student models with the semi-supervised pipeline described in section 9.2, in which unlabeled soundscapes are incorporated through teacher-generated pseudo-labels. As with the baselines, our guiding principle was to maximize diversity within the model pool: the configurations summarized in Table 7 vary across backbones, augmentation and Mixup regimes, teacher ensembles, number of epochs, and the training data used.

Two regimes are distinguished by the *Training data* column. Models trained on *All* data use the entire dataset, including the soundscapes otherwise held out for validation, whereas models trained on a single *Fold* retain the standard split. This distinction matters for evaluation. Unlike the baselines, the models reported here are single-fold models rather than five-fold ensembles, and their predictions are distilled from teachers trained across all folds. Because a teacher trained on a given fold may have seen soundscapes that overlap with another model's validation set, the local soundscape ROC-AUC can no longer be trusted: the pseudo-labels propagate information across the fold boundaries, resulting in indirect data leakage. We therefore rely exclusively on the public and private leaderboard scores to assess the models at this stage.

Even under this constraint, the semi-supervised models show a clear improvement over their supervised counterparts. The leaderboard scores cluster in a noticeably higher range than the baselines of Table 4, with private scores reaching up to 0.952 and most configurations exceeding 0.94 on both splits. The benefit is consistent across backbones and augmentation regimes, which indicates that the gain stems from the pseudo-labeling and consistency mechanism operating on genuine soundscape signals rather than from any single favorable configuration. As intended, the diversity across teachers, training data, and augmentation provides a heterogeneous pool of strong models that the ensembling stage described in section 13 can exploit.

We also tried to run multiple iterations of pseudo-labeling, but it didn't yield any gains, we suspect that this could be due to the confirmation bias we discussed in section 9.1.

Table 7
Models trained with the semi-supervised pipeline.

Fold	Baselines	Epochs	Training data	Backbone	Augmentation	Mixup α	Public	Private
4	3	32	All	tf_efficientnetv2_b3	extreme	1.0	0.947	0.950
0	3	20	All	tf_efficientnetv2_b3	light	2.0	0.947	0.951
0	4	20	All	tf_efficientnetv2_s	strong	0.5	0.942	0.951
0	2	32	All	tf_efficientnetv2_b0	strong	0.2	0.939	0.941
4	4	20	Fold	tf_efficientnetv2_s	strong	0.5	0.945	**0.952**
0	3	20	Fold	tf_efficientnetv2_b3	light	0.5	0.943	0.942
3	1	20	Fold	tf_efficientnetv2_b0	strong	0.5	**0.950**	0.937
1	4	20	Fold	tf_efficientnetv2_s	light	2.0	0.944	0.951

13. Ensembling and post-processing

13.1. Logits sharpening

Before averaging, the logits of each model are passed through a tempered sigmoid $\sigma(z/T)$ with $T = 0.8$ instead of the standard sigmoid. This sharpens the per-model probabilities, pushing confident predictions closer to the extremes before they are combined. The effect on the leaderboard is marginal, but it slightly improves the private score without harming the public one.

13.2. Sliding window

At inference, each one-minute soundscape is processed as twelve 5-second segments. To account for vocalizations that straddle two consecutive segments, we predict on overlapping windows with a 2.5 seconds stride. For each scored 5-second segment, we take the element-wise maximum over the segment and its two overlapping neighbors. This temporal max-pooling recovers calls that fall near a segment boundary and provides the single largest improvement among our post-processing steps, lifting both scores significantly.

13.3. Smoothing

We apply a class-dependent temporal smoothing that reflects the acoustic nature of each taxon. The predictions for species with short, punctual calls (Aves, Mammalia, Reptilia) are kept as they are while those with continuous or sustained calls (Insecta, Amphibia) are smoothed across neighboring segments. The smoothed predictions are then blended with the per-file maximum through a class-dependent global factor, so that taxa expected to call continuously throughout a recording are pulled toward their file-level activity while punctual species retain their temporal localization. This step further improves both scores and yields our best private result.

13.4. Ensembling

Our ensembling strategy is deliberately simple. Given the diverse pool of models described in section 12, we combine the predictions of the 8 models by a plain arithmetic mean over the per-segment class probabilities after the augmentations previously described.

13.5. Priors

Finally, the priors were computed by aggregating the predictions of the ensemble on the different available metadata (site, time of the day, and date). We pre-compute priors with the ensemble on all

the available unlabeled soundscapes and reduce the predictions to the maximum probability for each file. We combined these pre-computed priors with the predictions on the test data and the labeled soundscapes data. The granularity for the time of the day is per hour while the granularity for the date is per month. The priors are applied by performing a geometric averaging between the predictions obtained previously and the priors as described in equation 13. While conceptually appealing, this adjustment provided only a negligible gain on the public set and did not improve the private score, even slightly degrading it.

$$\hat{y}_p = \hat{y}^{0.8} \left(P_{site}^{0.4} \, P_{date}^{0.2} \, P_{time}^{0.4} \right)^{0.2} \tag{13}$$

where P are the priors.

13.6. Results

The successive contributions of the ensemble and of each post-processing step are reported in Table 8.

Table 8
Post-processing and ensembling evaluation

	Public	Private	Δ Public	Δ Private
Ensembling	0.948	0.941		
+ Logits sharpening	0.948	0.942	+0.000	+0.001
+ Sliding window	0.956	0.952	**+0.008**	**+0.010**
+ Smoothing	0.959	**0.956**	+0.003	+0.004
+ Priors	**0.960**	0.956	+0.001	-0.000

14. Conclusions

In this working note, we described our approach to the BirdCLEF+ 2026 competition, which achieved 12th place out of 4,094 teams. Our framework addresses two central challenges of the task, the noisy annotations and the domain gap between focal recordings and field soundscapes, through a single unifying idea: audio mixing is label-preserving, and predictions on mixed inputs should therefore remain consistent with the combined predictions on their unmixed sources. This principle underlies our Mixup consistency regularization term and extends naturally from the supervised setting to a semi-supervised one, where it operates jointly over labeled focal recordings and pseudo-labeled soundscapes.

Our central empirical finding is that the two mechanisms we introduced, the class-wise asymmetric loss (CASL) and the Mixup consistency term, contribute little when taken in isolation. Applied separately, each yields at best a marginal and inconsistent change in performance. It is only their combination that produces a pronounced and robust improvement, both locally and on the leaderboard. We interpret this as a genuine synergy rather than an additive effect. This interdependence was consistent across an alternative random seed, which supports its robustness.

The supervised baseline was subsequently extended into a semi-supervised pipeline that incorporates the large pool of unlabeled soundscapes through teacher-generated pseudo-labels within the same consistency framework, letting the model learn from genuine soundscape signals rather than on synthetic mixtures of focal recordings alone. A simple ensemble of diverse models, combined with a pre-training stage that yielded significant gains, and lightweight post-processing steps, in particular an overlapping sliding window and a class-dependent temporal smoothing, provided further consistent

gains on both the public and private leaderboards.

15. Discussion and future work

Our results were obtained under a number of deliberate simplifications, and we view several of them as promising directions for future work.

The first concerns the class-wise asymmetric loss. The hyperparameters were only tuned based on assumptions and the margin was ignored. The margin is precisely the component that hard-discards likely mislabeled negatives, and our motivation for a class-wise treatment, that the rate of missing positives varies strongly across taxa, applies just as naturally to it. A class-wise margin p_m, assigning a larger margin to taxa with a high suspected false-negative rate such as the unannotated insect backgrounds, would offer a more targeted mechanism to reject corrupted negative gradients than the soft attenuation provided by γ_- alone, and would let the two parameters act in complementary ways. We expect this to be especially relevant in combination with the consistency term, since L_C amplifies the model's own predictions and therefore benefits most from a loss that cleanly suppresses unreliable negatives.

More broadly, all of our γ values were set from qualitative domain assumptions rather than measured noise rates, as we lacked quantitative estimates of the per-class missing-label frequencies. These rates could be, for example, directly estimated from the data by comparing pseudo-labels on the focal recordings to their ground-truth.

A second direction concerns the semi-supervised stage. Additional iterations of pseudo-labeling did not yield further gains in our experiments, which we attribute to the confirmation bias inherent to training on a model's own predictions. Our current safeguards, a persistent supervised signal and a ramped consistency weight, mitigate but do not eliminate this effect. Mechanisms that more actively guard against it, such as confidence-based filtering of pseudo-labels, disagreement-based teacher selection, or a slowly updated teacher rather than a fixed one, may be needed to make iterated self-training profitable.

Finally, incorporating recording metadata via priors was largely unsuccessful, yielding only a negligible gain on the public set and even a small decline in performance on the private set. Nonetheless, the site, time, and date information carries genuine ecological information about potential migration dynamics, which we believe to be especially important in the Pantanal.

16. Declaration on Generative AI

During the preparation of this work, the authors used Gemini Pro from Google, and Claude Opus from Anthropic in order to: Abstract drafting, drafting content, formatting assistance, paraphrase and reword, grammar and spelling check, and peer review simulation.
Further, the author used the same models in order to code ad-hoc functions and code reviewing.
After using these tools, the author reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

[1] H. Goëau, S. Kahl, H. Glotin, R. Planqué, W.-P. Vellinga, A. Joly, Overview of birdclef 2018: monospecies vs. soundscape bird identification, in: CLEF 2018-conference and labs of the evaluation forum, volume 2125, 2018.

[2] S. Kahl, F.-R. Stöter, H. Goëau, H. Glotin, R. Planque, W.-P. Vellinga, A. Joly, Overview of birdclef 2019: large-scale bird recognition in soundscapes, in: CLEF 2019-conference and labs of the evaluation forum, volume 2380, CEUR, 2019.
[3] S. Kahl, M. Clapp, A. W. Hopping, H. Goëau, H. Glotin, R. Planqué, W.-P. Vellinga, A. Joly, Overview of birdclef 2020: Bird sound recognition in complex acoustic environments, in: Conference and Labs of the Evaluation Forum (CLEF 2020), 2696, CEUR-WS, 2020, pp. 13–p.
[4] S. Kahl, T. Denton, H. Klinck, H. Glotin, H. Goëau, Overview of birdclef 2021: Bird call identification in soundscape recordings, CEUR-WS, 2021.
[5] S. Kahl, A. Navine, T. Denton, H. Klinck, P. Hart, H. Glotin, H. Goëau, W.-P. Vellinga, R. Planqué, A. Joly, Overview of birdclef 2022: Endangered bird species recognition in soundscape recordings, CEUR-WS, 2022.
[6] S. Kahl, T. Denton, H. Klinck, H. Reers, F. Cherutich, H. Glotin, H. Goëau, W.-P. Vellinga, R. Planqué, A. Joly, Overview of birdclef 2023: Automated bird species identification in eastern africa, in: Conference and Labs of the Evaluation Forum (CLEF-WN 2023), volume 3497, CEUR-WS, 2023, pp. 1934–1942.
[7] S. Kahl, T. Denton, H. Klinck, V. Ramesh, V. Joshi, M. Srivathsa, A. Anand, C. Arvind, H. Cp, S. Sawant, et al., Overview of birdclef 2024: Acoustic identification of under-studied bird species in the western ghats, in: Conference and Labs of the Evaluation Forum (CLEF 2024), CEUR-WS, 2024, pp. 1948–1957.
[8] 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, et al., Overview of birdclef+ 2025: Multi-taxonomic sound identification in the middle magdalena, colombia, CEUR-WS, 2025.
[9] S. Kahl, T. Denton, L. Sugai, L. Piatti, W. H. Arruda Moreno, M. M. Barbosa, M. Eibl, C. Z. Fieker, C. M. Garcia, D. L. Hokama Sousa, J. E. d. A. Júnior, R. C. Kridler, M. Lasseck, A. V. d. Melo, H. Müller, M. d. O. Neves, M. G. d. Reis, L. K. Sampaio Teles, K.-L. Schuchmann, J. L. M. M. Sugai, K. T. P. Azevedo, P. d. N. Lopes, M. I. Marques, H. Klinck, H. Glotin, H. Goëau, W.-P. Vellinga, R. Planqué, A. 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.
[10] 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, 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.
[11] T. Ridnik, E. Ben-Baruch, N. Zamir, A. Noy, I. Friedman, M. Protter, L. Zelnik-Manor, Asymmetric loss for multi-label classification, in: Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 82–91.
[12] S. Kahl, C. M. Wood, M. Eibl, H. Klinck, Birdnet: A deep learning solution for avian diversity monitoring, Ecological Informatics 61 (2021) 101236.
[13] B. Ghani, T. Denton, S. Kahl, H. Klinck, Global birdsong embeddings enable superior transfer learning for bioacoustic classification, Scientific Reports 13 (2023) 22876.
[14] B. van Merriënboer, V. Dumoulin, J. Hamer, L. Harrell, A. Burns, T. Denton, Perch 2.0: The bittern lesson for bioacoustics, arXiv preprint arXiv:2508.04665 (2025).
[15] H. Zhang, M. Cisse, Y. N. Dauphin, D. Lopez-Paz, mixup: Beyond empirical risk minimization, arXiv preprint arXiv:1710.09412 (2017).
[16] A. Tarvainen, H. Valpola, Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results, Advances in neural information processing systems 30 (2017).
[17] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. Raffel, E. D. Cubuk, A. Kurakin, C.-L. Li, Fixmatch: Simplifying semi-supervised learning with consistency and confidence, Advances in neural information processing systems 33 (2020) 596–608.
[18] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, C. Raffel, Mixmatch: A holistic

approach to semi-supervised learning, Advances in neural information processing systems 32 (2019).
[19] V. Verma, K. Kawaguchi, A. Lamb, J. Kannala, A. Solin, Y. Bengio, D. Lopez-Paz, Interpolation consistency training for semi-supervised learning, Neural Networks 145 (2022) 90–106.
[20] Xeno-canto - Bird sounds from around the world, Xeno-canto - bird sounds from around the world, https://xeno-canto.org, 2026. Accessed: 2026-06-16.
[21] iNaturalist, inaturalist, https://www.inaturalist.org, 2026. Accessed: 2026-06-16.
[22] V. Sydorskyi, F. Gonçalves, Tackling domain shift in bird audio classification via transfer learning and semi-supervised distillation: A case study on birdclef+ 2025, CLEF Working Notes (2025) 09–12.
[23] M. Tan, Q. Le, Efficientnetv2: Smaller models and faster training, in: International conference on machine learning, PMLR, 2021, pp. 10096–10106.
[24] A. L. Maas, A. Y. Hannun, A. Y. Ng, et al., Rectifier nonlinearities improve neural network acoustic models, in: Proc. icml, volume 30, Atlanta, GA, 2013, p. 3.
[25] R. Wightman, Pytorch image models, https://github.com/huggingface/pytorch-image-models, 2019. doi:10.5281/zenodo.4414861.
[26] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, Q. V. Le, Specaugment: A simple data augmentation method for automatic speech recognition, arXiv preprint arXiv:1904.08779 (2019).
[27] K. Xu, D. Feng, H. Mi, B. Zhu, D. Wang, L. Zhang, H. Cai, S. Liu, Mixup-based acoustic scene classification using multi-channel convolutional neural network, in: Pacific Rim conference on multimedia, Springer, 2018, pp. 14–23.
[28] Q. Xie, M.-T. Luong, E. Hovy, Q. V. Le, Self-training with noisy student improves imagenet classification, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 10687–10698.
[29] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, arXiv preprint arXiv:1711.05101 (2017).
[30] O. R. developers, Onnx runtime, https://onnxruntime.ai/, 2021. Version: 1.26.0.

A. Training setup

The models are trained with an AdamW[29] optimizer with a cosine annealing scheduler with learning rates ranging from 5×10^{-4} to 10^{-6} (10^{-8} for the pre-training models). The mel spectrograms were normalized between 0 and 1 based on recording-wise minimum and maximum values. The mel spectrogram configuration can be found in Table 9.
Unless otherwise specified, models are trained with a batch size of 32. After training completes, the final weights for each model are obtained by averaging the weights of the last three epochs. All the reported leaderboard scores include the application of the all the post-processing steps before the priors except for the scores reported in Table 8.

Table 9
Mel spectrogram configuration

Parameter	Value
Sample rate	32,000
n_mels	160 / 224
Frequency range	20/50 - 16,000
n_fft	2048
hop_length	512

B. Inference time

The models were compiled in ONNX[30] format, and the inference time depended solely on the backbone architecture and the dimensions of the mel spectrograms. The inference times for the various configurations in CPU-only Kaggle notebooks are summarized in Table 10.

Table 10
Inference duration for 600 recordings for the different configurations

Backbone	n_mel	Inference time [minutes]
tf_efficientnetv2_b0	224	6:40
tf_efficientnetv2_b3	224	14:21
tf_efficientnetv2_s	160	12:19

C. Additional ablation experiments

Table 11
5-folds baseline loss function evaluation with an alternative seed

Loss	L_C	Soundscape ROC-AUC	Public	Private
BCE	No	0.892 ± 0.033	0.926	0.928
BCE	Yes	0.898 ± 0.021	0.921	0.928
CASL	No	0.892 ± 0.031	0.920	0.935
CASL	Yes	$\mathbf{0.909 \pm 0.028}$	**0.933**	**0.942**

Table 12
5-folds baseline loss function evaluation.

Loss	Soundscape ROC-AUC	Public	Private
BCE	0.894 ± 0.033	0.924	0.932
ASL ($\gamma_- = 2, \gamma_+ = 0, p_m = 0.05$)	0.899 ± 0.030	0.927	0.924
ASL ($\gamma_- = 4, \gamma_+ = 1, p_m = 0.05$)	0.911 ± 0.040	0.924	0.923
CASL	$\mathbf{0.918 \pm 0.024}$	**0.938**	**0.940**

Table 13
5-folds baseline loss function evaluation for CASL class ablation. All γ_- and γ_+ parameters are set to 0, except for the γ_- of the corresponding class, which is set to 2.

Loss	Class	Soundscape ROC-AUC	Public	Private
CASL	Aves	0.908 ± 0.025	0.922	0.933
CASL	Amphibia	0.901 ± 0.039	0.930	0.936
CASL	Insecta	$\mathbf{0.920 \pm 0.034}$	**0.935**	**0.941**
CASL	Mammalia	0.903 ± 0.030	0.935	0.937
CASL	Reptilia	0.912 ± 0.021	0.932	0.936

D. Notation

Table 14
L_C scheduling evaluation

L_C scheduling	Soundscape ROC-AUC	Public	Private
	0.898 ± 0.036	0.925	0.931
L_C	0.912 ± 0.028	0.935	0.938
$\eta\, L_C$	0.913 ± 0.036	0.932	0.936
$(0.5 + \eta)L_C$	$\mathbf{0.918 \pm 0.024}$	**0.938**	**0.940**

Table 15
Pseudo-labeling loss function evaluation on a semi-supervised pipeline training on fold 0 with the baselines from Table 5 as teachers.

Loss	L_C (Baseline)	L_C (Semi-supervised)	Public	Private
BCE	No	No	0.921	0.939
BCE	No	Yes	0.923	0.936
BCE	Yes	No	0.918	0.934
BCE	Yes	Yes	0.919	0.931
CASL	No	No	0.930	0.939
CASL	No	Yes	0.932	0.944
CASL	Yes	No	0.940	0.946
CASL	Yes	Yes	**0.943**	**0.947**

Table 16
Summary of notation

Symbol	Description
$x,\ y$	Batch of 5-second input signals and their multi-label targets
$x_l,\ y_l$	Labeled (focal) inputs and ground-truth targets
$x_u,\ \hat{y}_u$	Unlabeled soundscape inputs and teacher-generated pseudo-labels
Model*	Full forward operator: signal $\rightarrow$ log-mel $\rightarrow$ augmentation $\rightarrow$ logits
Teacher$^*_k$	Full forward operator of the k-th teacher model: signal $\rightarrow$ log-mel $\rightarrow$ augmentation $\rightarrow$ logits
$\hat{y}$	Model prediction on an individual input
$\tilde{x}$	Mixed input, $\lambda x_i + (1-\lambda)x_j$
$\tilde{y}$	Mixed label, $\max(y_i, y_j)$
$\hat{\tilde{y}}$	Prediction on the mixed input, Model$^*(\tilde{x})$
$\tilde{\hat{y}}$	Max of individual predictions, $\max(\hat{y}_i, \hat{y}_j)$; stop-gradient target for L_C
p	Predicted probability for a class (CASL)
p_m	Shifted probability, $\max(p - m,\ 0)$
$\gamma_+,\ \gamma_-$	Positive / negative focusing parameters of (C)ASL
λ	Mixup interpolation coefficient, $\lambda \sim \text{Beta}(\alpha, \alpha)$
α	Beta-distribution parameter controlling Mixup strength
$L_S,\ L_M,\ L_C$	Supervised, mixed-supervised, and consistency loss terms
η	Consistency ramp coefficient, $\eta = \text{epoch}/\#\text{epochs}$
K	Number of teacher models in the pseudo-labeling ensemble / Number of folds
T	Temperature (pseudo-label sharpening; tempered sigmoid)
$w_i,\ c_i$	Sampling weight and recording count for species i
β	Sampling exponent (set to 0.8)
$P_{site},\ P_{date},\ P_{time}$	Metadata-based priors