



Research Code Competition

BirdCLEF 2021 - Birdcall Identification

Identify bird calls in soundscape recordings



\$5,000

Prize Money



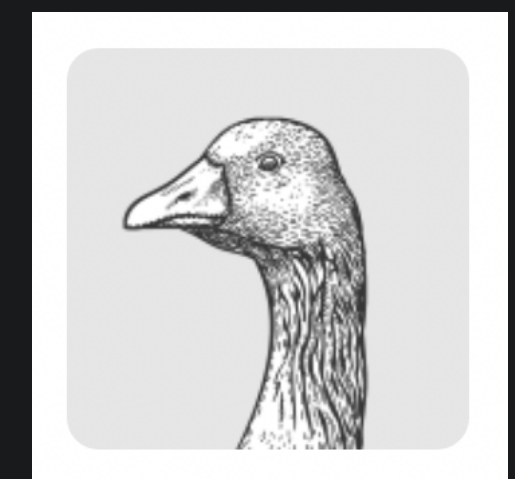
Cornell Lab of Ornithology · 816 teams · 6 days ago

BirdCLEF2021まとめ

コンペ概要と上位者解法

[English version](#) also available

[start](#)
([@startjapan](#))



(Speaker Deckの概要欄から各リンクに飛べます)

(アイコン探し中...)

自己紹介



start

Med Student at Hiroshima Univ.
Hiroshima, Hiroshima, Japan
Joined a year ago · last seen in the past day

Followers 5
Following 6

**Competitions Master**

[Home](#) [Competitions \(12\)](#) [Datasets \(3\)](#) [Code \(1\)](#) [Discussion \(1\)](#) [Followers \(5\)](#) [Notifications](#) [Account](#) [Edit Public Profile](#)

Competitions Master

Rank **98**
of 161,854

1

3

0

BirdCLEF 2021 - ...
a day ago
Top 1%

1st
of 816

Shopee - Price ...
23 days ago
Top 2%

42nd
of 2426

Mechanisms of ...
6 months ago
Top 5%

187th
of 4373

Datasets Contributor

Unranked

0

0

0

cassava_diseas...
4 months ago

1 vote

pytorch_image_...
a month ago

0 votes

cassava-diseas...
4 months ago

0 votes

Notebooks Contributor

Unranked

0

0

0

diamond_for_sh...
5 months ago

1 vote

Discussion Contributor

Unranked

0

0

1

1st Place Solution
3 months ago

1 vote

Bio

Edit

Med Student at Hiroshima Univ.

自己紹介

広島の医学生.

国試の勉強の傍ら医療ベンチャー (株)MNES にてインターンをしており, 同ベンチャーが後援する [LAIME](#) という学生向け機械学習勉強サークルで研鑽を積んでいます.

本コンペで優勝し, Masterとなりました.
(多分にチームメイトのおかげですが...)

あと, 良いアイコンを探しています.

kaggle : [@startjapan](#)

twitter : [@startjapanml](#)

コンペ概要

コンペ概要

- 5秒刻みの音声セグメントから鳴いている鳥種を特定するコンペ
(2020年にも同じ主催者が類似コンペを開催している → 2020年の鳥コンペと呼びます)
- trainデータはxeno-cantoという鳥の鳴き声共有サイトから取得された音声
(train_short_audio)
- testデータは10分×80件の音声ファイル, これを5秒ごとに区切り予測する
(test_soundscapes)
- 上記とは別にvalidation用の音声(10分×20件)も与えられた
(train_soundscapes)

Data Explorer

39.28 GB

- ▶  test_soundscape
- ▶  train_short_audio
- ▶  train_soundscape
-  **sample_submission.csv**
-  test.csv
-  train_metadata.csv
-  train_soundscape_labels...

[test_soundscape]

testデータ. 10分×80件だが提出しないとアクセスできない.
トータル4つの場所で録音されている.

[train_short_audio]

学習データ. 鳥種ごとに音声が集められている.
合計で62874件の音声データ.

[train_soundscape]

test_soundscapeに近い音響ドメインを持つ.
10分×20件あり, test_soundscapeを録音した4つの場所の
うち2つの場所で録られた音声

[train_metadata.csv]

train_short_audioに対するmetadata. shapeは(62784, 14)

[train_soundscape_labels.csv / test.csv]

10分のファイルを5秒セグメントに分けた際の骨組みを提供.
train_soundscape_labels.csvはtrain_soundscapeに,
test.csvはtest_soundscapeに対応し,
前者にのみ正解ラベルが付いている.

提出形式 & 評価指標

- 1つのセグメントに対して複数の鳥種を予測として提出可能
- 鳥が鳴いていないセグメントに対しては"nocall"という文字列を提出
- 評価指標は行ごとのmicro-F1スコアの平均

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{\text{TP}}{\text{TP} + \frac{1}{2}(\text{FP} + \text{FN})}$$

2020年の鳥コンペとの差異

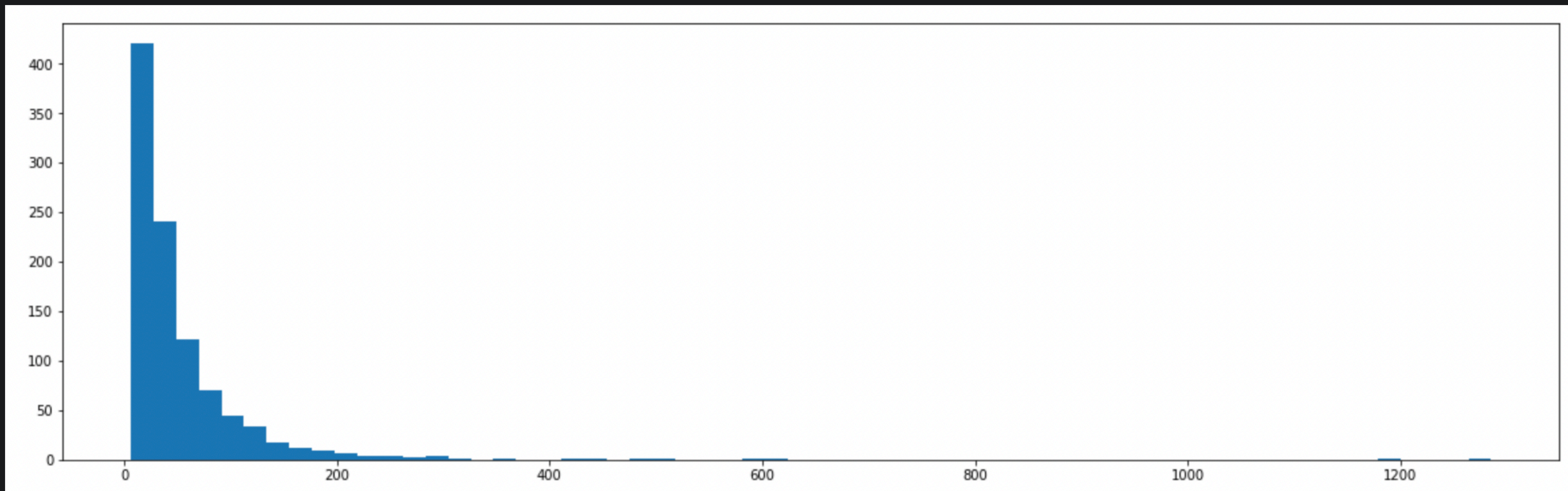
2020年の鳥コンペとの差異

- train_soundscapesの存在
 - train_short_audioとtestデータの間には音響ドメインの差が大きい
 - 今回のコンペではよりtestデータに近い音響ドメインを持つtrain_soundscapesが与えられた
 - validation用途で用いられることが多かったが、中には工夫して学習に用いる人もいた
- testデータの位置情報にアクセスできた
 - testデータの各ファイル名に場所の情報が入っていることは保証されていた (日付も)
 - 従って、これらの情報も何らかの形でパイプラインに組み込む必要があった

(参考 : [Starter and some thoughts](#) by @hidehisaarai1213)

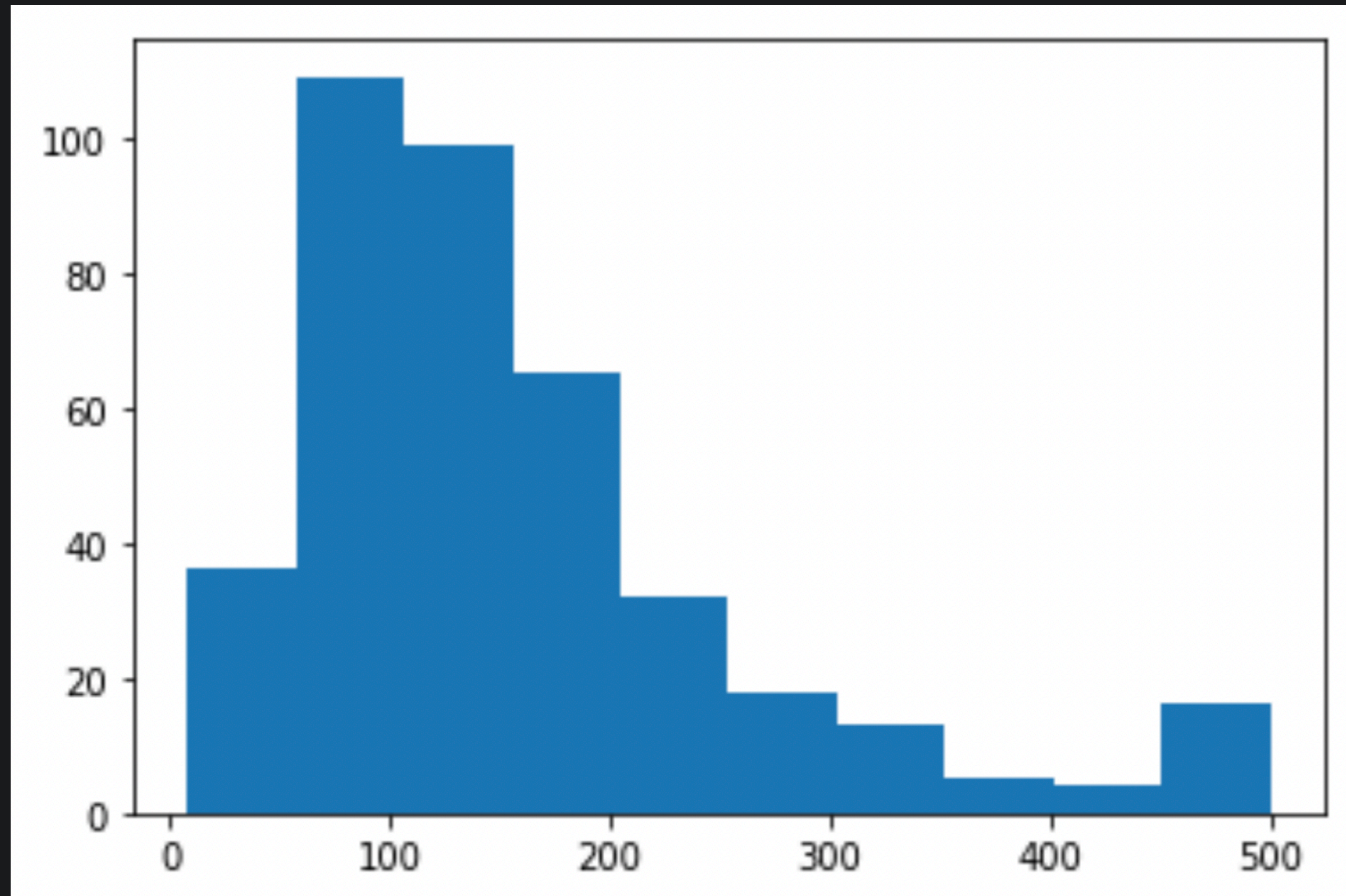
EDA (train_short_audio編)

1ファイルあたりの音声の長さ (train_short_audio)



- ※ train_short_audioのうち1000件 (/ 62874件) の音声ファイルをランダムサンプリング
- ※ 横軸：1ファイルあたりの音声の長さ [秒]
- ※ 縦軸：度数 (合計1000件)

1種の鳥に対して何件の音声ファイルがある？ (train_short_audio)



※ 横軸：各鳥における音声ファイル数 (train_short_audio内)

※ 縦軸：度数 (合計397種)

secondary labelsには欠損があると明記されている (train_short_audio)

2. Background species

The metadata for each recording lists the number of audible background species. The data field “*seconday_labels*” contains lists of eBird codes (i.e., primary labels) that recordists annotated. It is important to note that these lists might be incomplete, and you might be able to hear background species, although none are specified in the metadata. Therefore, lists of secondary labels are not very reliable, but they might still be useful for multi-label training (e.g., through loss masking for background species).

([BirdCLEF2021: Exploring the data](#)より引用)

EDA (soundscapes編)

全行nocall提出でPublicLBのnocall率は分かる

BirdCLEF2021

Private

Public

[CLEF\[1st sub\]](#)

(version 2/2)

a month ago by [start](#)

全行nocall

Succeeded

0.4799

0.5467

(参考：2020年の鳥コンペ)

Private

Public

[PreviousCLEF\[all_rows_nocall\]](#)

(version 3/4)

a month ago by [start](#)

過去コンペ全行nocall

Succeeded 

0.577

0.544

一方でtrain_soundscapeではやや高いnocall率

[5]:

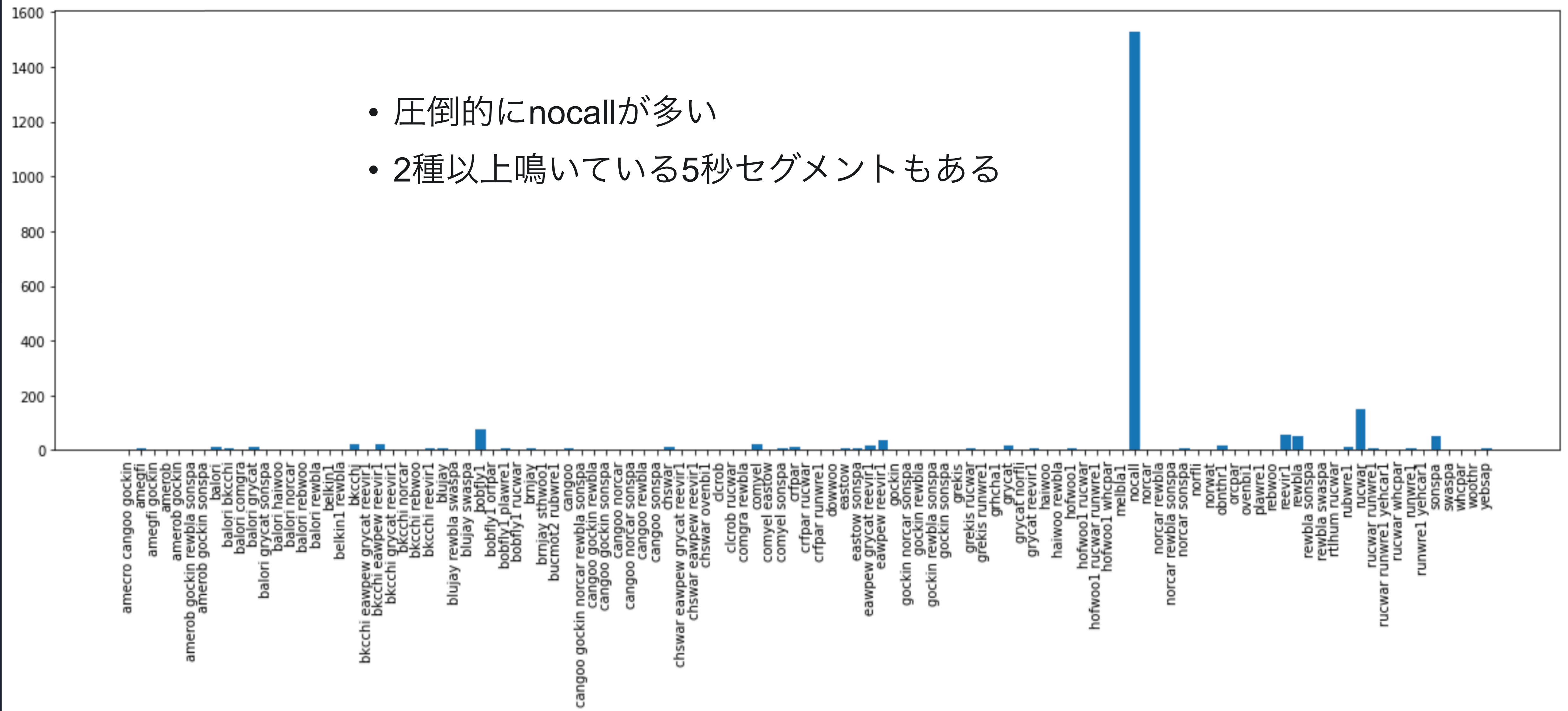
```
import pandas as pd

train_soundscape = pd.read_csv('../input/birdclef-2021/train_soundscape_labels.csv')
nocall = train_soundscape.query('birds=="nocall"')
print(len(nocall)/len(train_soundscape))
```

0.6370833333333333

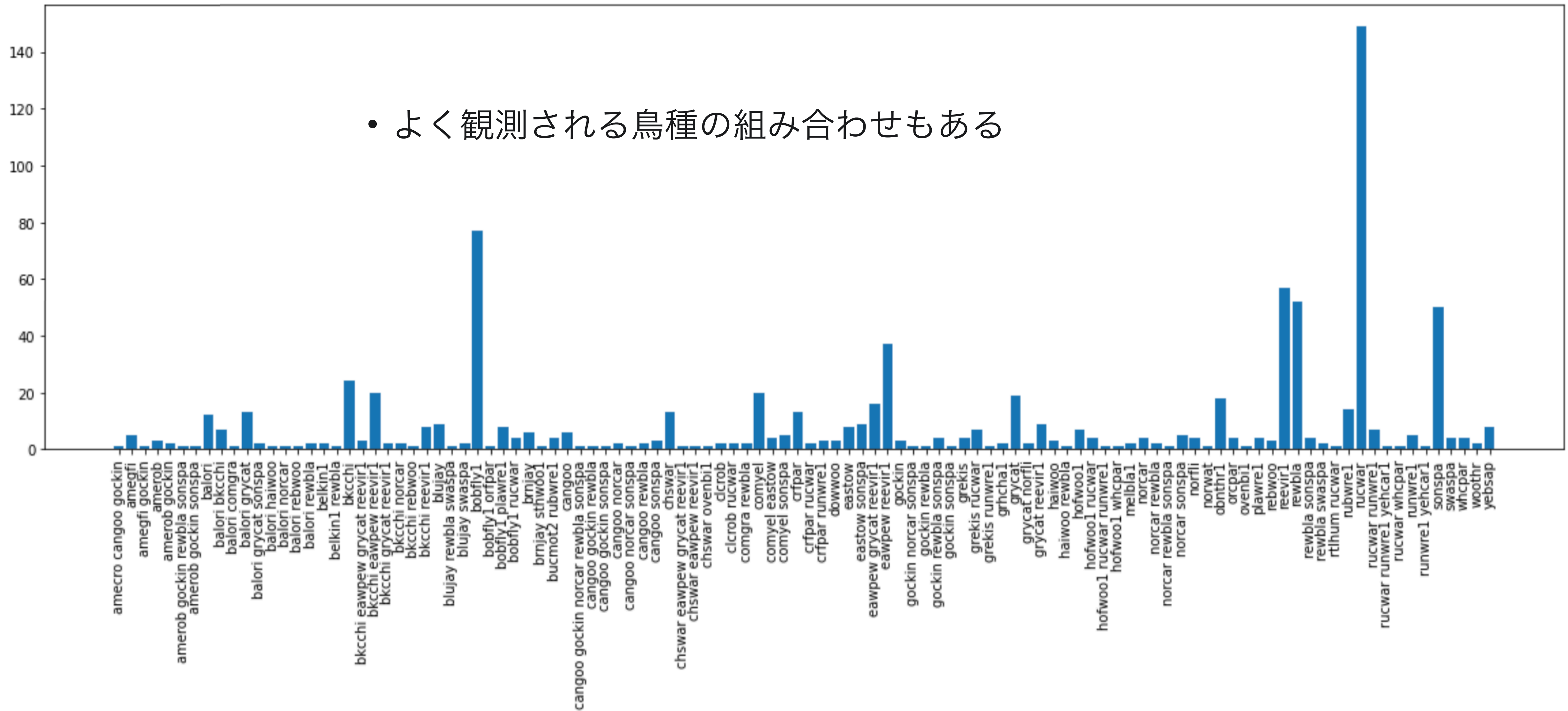
train_soundscapeにおける目的変数の分布 (nocall込み)

- 圧倒的にnocallが多い
- 2種以上鳴いている5秒セグメントもある

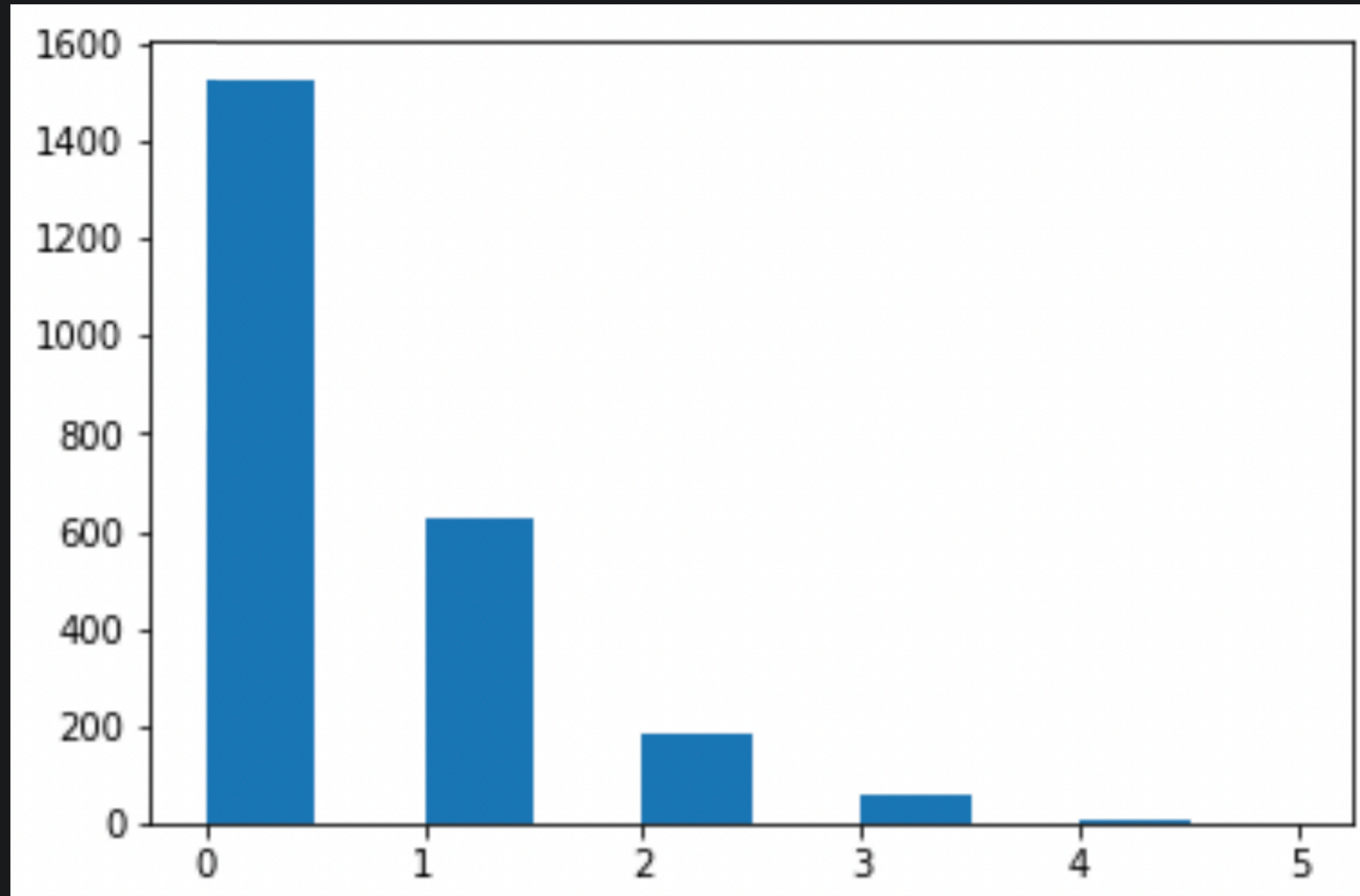


train_soundscapeにおける目的変数の分布 (nocall削除版)

- よく観測される鳥種の組み合わせもある



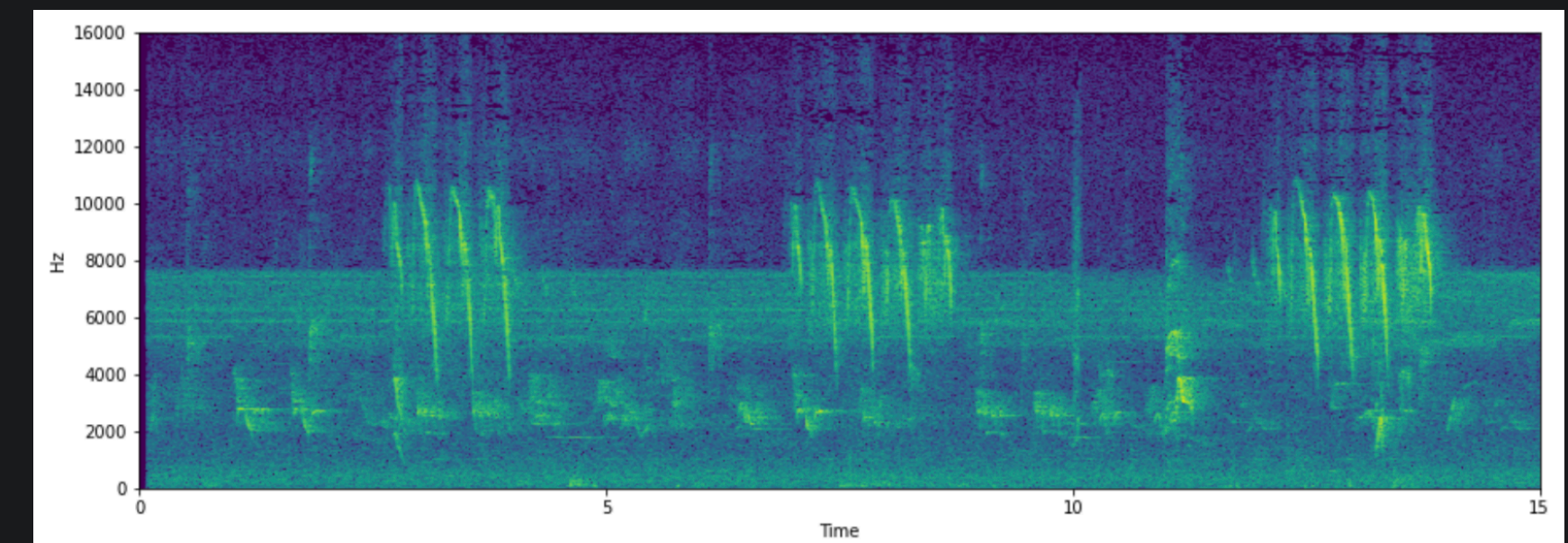
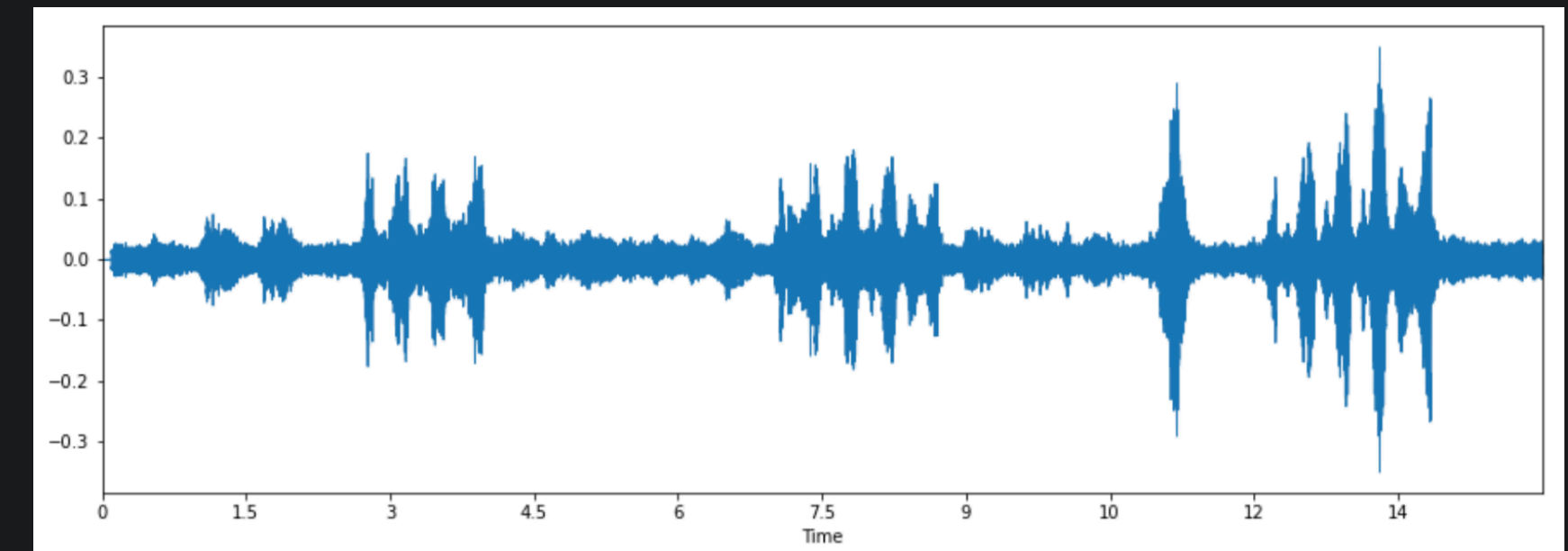
train_soundscapeにおいて5秒セグメント内で同時に鳴いている鳥の数



音声認識タスクのベーシックな解法

音声認識タスクのベーシックな解法

音声データは横軸が時間、縦軸が周波数、
各ピクセルが信号成分の強度を示す画像
(スペクトログラム) に変換可能であり、
これに対してCNNなどを適応すると
従来通りの画像処理として扱える。



CNN

※ 本コンペでは縦軸(周波数)にメル尺度を使用したメルスペクトログラムがよく使われた

※ メル尺度とは：音の周波数に関して、この尺度上での差が同じであれば人間が耳で感じる音の高さの差も同じになる

(画像は[BirdCLEF2021: Processing audio data](#)より引用)

本コンペ特有のクセ

本コンペ特有のクセ

- train_short_audioに対してweak labelしか振られていない (**weak label問題**)
 - 数十秒の音声データ全体に対してラベルが付与されている
 - 5秒区切りのセグメントレベルでどの鳥が鳴いているかが分からない
- train_short_audioの一部でラベルの欠損がある (**noisy label問題**)
 - 特にsecondary_labels(*)には欠損があると明記されている
- 録音日や場所の情報などのmetadataも何らかの形で織り込む必要がある (**metadataの織込**)
- 予測対象の前後のセグメントで鳥が鳴いているかという情報も意味を持つ可能性がある (**セグメント前後情報の織込**)
- train_soundscapesとtest_soundscapesでnocall率が大きく異なる (**CV戦略確立の困難**)

※ train_short_audioのラベルにはprimary_labelとsecondary_labelsの2種類がある

上位者の解法

[top solutions and approaches](#)

上記のdiscussionに上位者解法へのリンクがまとまっています

1st place (ours!)

[\[1st Place\] Quick Solution](#)

[\[1st Place\] Detailed Solution](#)

tl;dr

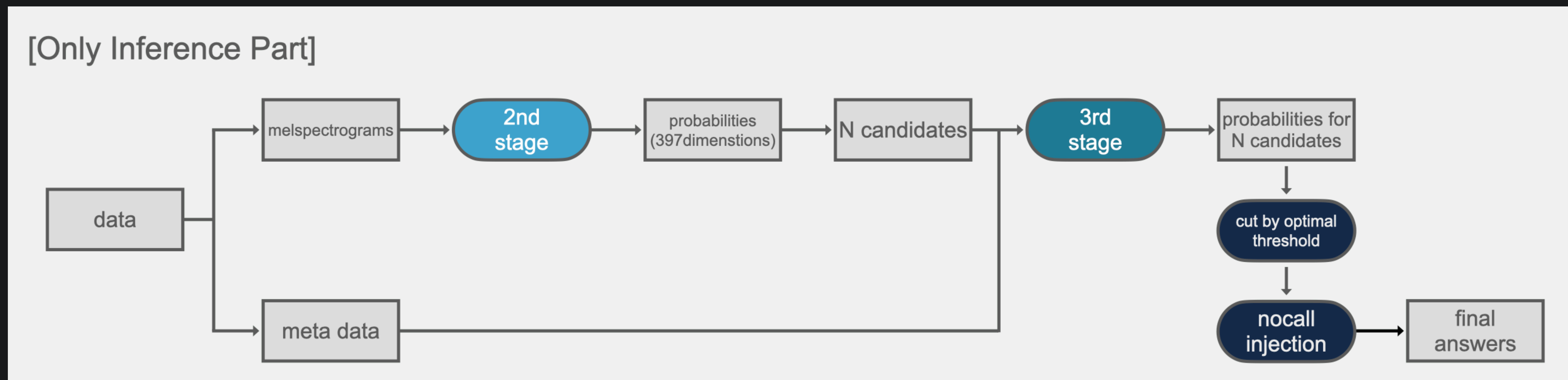
1st stage : 外部データ(freefield1010)を使ってbinary nocall detector 作成 (1 : 何か鳥鳴いてる / 0 : nocall)

2nd stage : nocall detectorを使ってtrain_short_audioからnocall部分のweightを減らした上で397次元マルチラベル分類器を作成

3rd stage : nocall detectorの結果, metadata, 2nd stageの結果などから自前table competitionを作成

最終的に自前table competitionにすることで

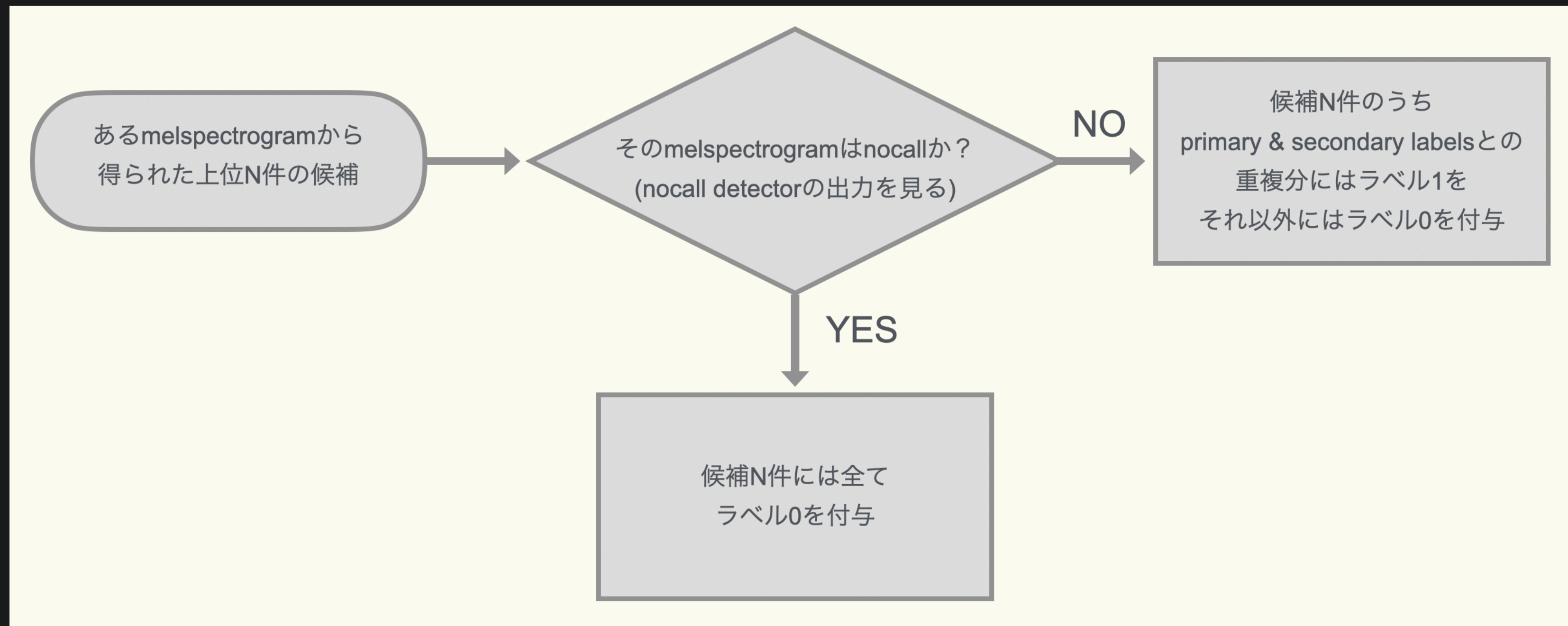
weak label問題, noisy label問題, metadataの織込, セグメント前後情報の織込などをまるっと解決！！



※ Inference Part のみの概略であり, 1st stage部分は省略しています

なぜtable化でweak label問題 & noisy label問題が解決される？

- 3rd stageのtarget変数 (0 : はずれ行 / 1 : あたり行) は以下の流れで決定される



- 数十秒の音声データに対して付与されたprimary & secondary labelsに対してセグメント単位で予測値を出せる nocall detectorとマルチラベル分類器の出力を組み合わせることでweak label問題を解決
- 仮にsecondary labelsに欠損があるとラベル0が付与されるがラベル0のサンプル数は比較的多く noiseはいい感じに埋もれる (noisy label問題の緩和)

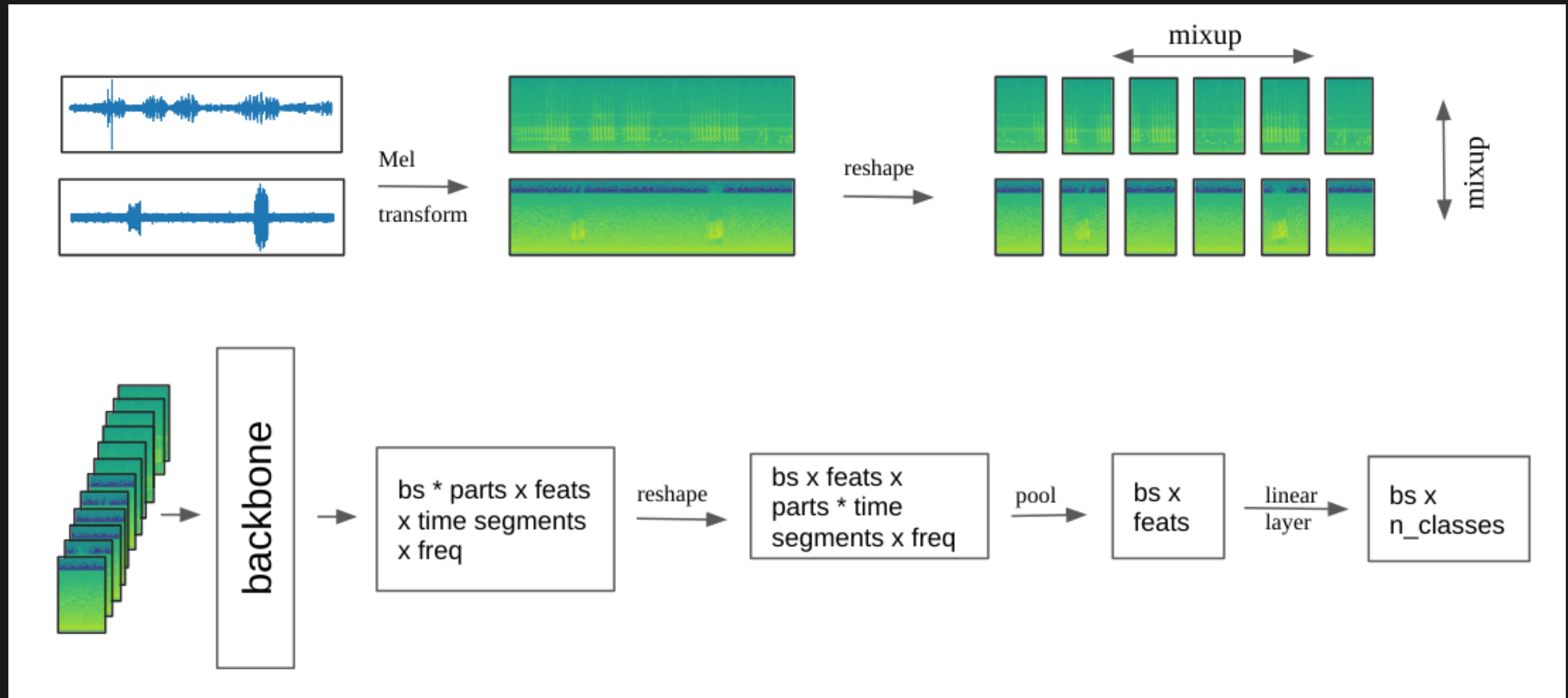
more details...

- チームメイトのkamiさん (twitter : [@634kami](#) / kaggle : [@kami634](#)) が以下に日本語で解法をまとめてくれました
[Kaggle の鳥コンペで1位を取った話 : BirdCLEF 2021 優勝解法](#)

2nd place

[2nd place solution](#)

2nd place



([2nd place solution](#)より引用)

2nd place

- train_short_audioから30秒単位で抽出したのち、5秒ごとに区切り、mixupする
(**weak label問題**への対応)
- train_soundscapesのうち10分間全く鳥が鳴かない音声ファイル3つの除外 & ブートストラップサンプリング
(**ロバストなCV戦略**)
- label smoothing & metadata中のrating列を用いて重み付け (**noisy label問題**への対応)
- 閾値選択の際のtips
 - LBではCVよりnocall率が低いので閾値を下げて鳥を多く予測
 - モデル間で確率値の分布が異なるため単一の確率値を閾値とするのはナンセンス
よって、パーセンタイルベースの閾値を使用
- その他 (後処理)
 - 鳥ごとの平均予測確率から個々の確率値を修正
 - **前後セグメント情報**を使用
 - nocall detectorの結果を加味
 - 時と場所の情報からありえない鳥種を予測をしている場合は削除 (**metadataの織込**)

4th place

[4th place solution](#)

4th place

- SEDモデルを使用, 入力は10-30秒 (**weak label問題**への対応)
(参考: [Introduction to Sound Event Detection](#) by @hidehisaarai1213)
- この方もmixupを使用
- pseudo labelingを実行 (**noisy label問題**への対応)
- 予測対象の5秒セグメントとそれを中心とする30秒セグメントのそれぞれに対するSEDの出力を組み合わせて最終出力とした (**セグメント前後情報の織込**)
 - 5秒セグメントに対しては小さな閾値, 30秒セグメントに対しては大きな閾値を使用
- 2位の解法と同様に, 時と場所の情報から観測される可能性が低いと判断した鳥種は削除 (**metadataの織込**)

5th place

[5th place solution](#)

5th place

- 2020年の鳥コンペで2位だった方であり，今回もそれをベースとしていた
- 前回からの改善点：SEDに変更で +1% (※1) / 閾値調整により +1% / アンサンブル方法改良で +1% (+ 地域情報をもとに予測ラベルの絞り込みもしたみたい(※2))
- augmentationが特徴的：画像を0.5-3乗 / n倍速 / 雨や会話などの音を追加 / ノイズ追加 / 0.5の確率で周波数調整 (1-4位はmixupやノイズ追加に留まる印象)
- primary labelはラベル1, secondary labelsはラベル0.3を付与
- 1つのセグメントで観測された鳥は10分の音声ファイル全体で拾い上げやすくなるよう調整 (※3)

※1 : weak label問題への対応

※2 : metadataの織込

※3 : セグメント前後情報の織込

8th place

[8th place writeup](#)

8th place

- 2020年の鳥コンペで6位だった方, 今回もSEDを使用 (**weak label問題**への対応)
- 学習時は5秒 or 20秒セグメント, 推論時は40秒セグメントを使用, 長いほうがよかった
また, 推論は0-40秒で行った次に20-60秒という風にoverlapをもたせた (**セグメント前後情報の織込**)
- augmentation: ガウシアンノイズ, ピンクノイズ, ボリューム調整, ピッチシフト
(mixupも上手く行ったが計算資源の問題で最終提出には組み込めなかったそう)
- 損失関数が特徴的 (BCEFocal2WayLoss)
- primary labelとsecondary labelsは同じように扱った
- pseudo labelingを実行 (**noisy label問題**への対応)
- 閾値はcall thresholdとnocall thresholdの2つが存在し, call thresholdを超えた鳥種は陽性とする一方で
全ての鳥種においてnocall thresholdを超えなかったセグメントにはnocallも付与 (鳥ラベルとnocallが共存しうる)
- 地域情報から存在するはずがない鳥種は予測していても除外 (**metadataの織込**)
- 鳥が鳴いている行とnocall行に分けてF1スコアを算出し $0.54 * \text{nocall_f1} + 0.46 * \text{call_f1}$ でCVを導出 (**ロバストなCV戦略**)

9th place

[9th Place solution](#)

9th place

- 学習時の入力は5-7秒セグメント
- secondary labelsの重みは小さくした
- mixup使用
- 可能な限りの多様性をもたせた
 - 時間分解能の異なるmel-spectrogram, hop_lengthは200と320
 - 様々なbackbone
 - augmentation : white noise, pink noise, band noise, nocall clips, mel-spectrogram画像の累乗
- 後処理
 - 10分の音声データ全体における各鳥の鳴く最大確率 or 平均確率で後処理 (セグメント前後情報の織込)
 - 地域情報からどれだけ各鳥が鳴く可能性があるか評価してその結果で後処理 (metadataの織込)
 - 1日の間で各鳥が鳴く最大確率を使って後処理 (metadataの織込)
- Squeeze width of test soundscapes by 2-5% (mostly to reverse far field effects)
- この方も閾値を2つ(call, nocall)設定し, 鳥ラベルとnocallの共存を認めた

11th place

[My journey \(11th solution\)](#)

11th place

- Public LBで長いこと首位を独占されていたCPMPさん
- 2020年の鳥コンペで18位, Rainforestコンペで11位をとられた方であり両者の解法をmixしたものをベースとしたそう
 - ー 2020年の鳥コンペの解法 : [18th place solution: efficientnet b3](#)
 - ー Rainforestコンペの解法 : [11th place, The 0.931 Magic Explained: Image Classification](#)
- 8位の解法と同じく $0.54 * \text{nocall_f1} + 0.46 * \text{call_f1}$ にてCVを算出 (ロバストなCV戦略)
- アンサンブルで確実にスコアが上昇すると過信しておりコンペ終了数日前までアンサンブルverを提出しなかったことを後悔なさっている (実際にはアンサンブルが効かなかったそう)

ありがとうございました！