

r-ribbit

Bioacoustics tool to detect pulsed vocalizations in audio recordings (R implementation)

The Repeat Interval Based Bioacoustic Identification Tool (RIBBIT) is an open-source tool designed to detect pulsing or periodic vocalizations of anurans (frogs and toads) or other vocal taxa in audio recordings. A Python implementation of RIBBIT is included in the open-source bioacoustics toolkit OpenSoundscape. We provide an R notebook that performs the same analysis for those more comfortable using R.

set up

Download the R markdown (.Rmd) notebook file and open it in RStudio, or clone this repository.

No installation is required, as we simply provide an R markdown notebook containing the RIBBIT implementation. The notebook depends on 3 packages, which the R notebook will attempt to install when you run it. We have tested the notebook with the following versions of each package:

```
versions 0.3 (used to install specific versions of packages)
base 3.6.1
utils 3.6.1
seewave 2.1.6
bspec 1.5
tuneR 1.3.3
```

basic use

To use this tool, you will need to - specify the location of the audio files you wish to analyze - select parameters for RIBBIT analysis

1. Specify location of audio files to analyze

Change the value of the variable `audio_folder` to the location of your audio files on your computer. For example, if I have .mp3 or .wav files in my Downloads folder on my mac, in a folder called `field_audio`, I would change line 30 to:

```
audio_folder <- '~/Downloads/field_audio'
```

If you wish to analyze audio contained in sub-folders within that folder, set

```
recursive <- TRUE
```

If you only wish to analyze audio files on the top level of that folder, leave it as `recursive <- FALSE`

2. Specify parameters for RIBBIT analysis

Analyzing files with RIBBIT requires that we specify frequency ranges related to the target vocalization and background noise, ranges of pulse repetition rates for the vocalization, and a few other parameters. See the comments in the notebook or the documentation of RIBBIT in OpenSoundscape for detailed instructions on how to select parameters for your target vocalization.

Change the values in lines 45-81 of the .Rmd file to reflect your choices of model parameters.

If you wish to save a .csv file (table) of top RIBBIT scores per file, be sure to use `save_max_scores_to_file = TRUE`. The resulting scores will be saved in the same location as your audio files.

3. Run the analysis

Run all cells (in RStudio, Run -> Run All) or run each cell individually and wait for the analysis to complete. If you have large files or many files, it may take a while. If your audio data is many gigabytes or terabytes, consider using the python implementation of RIBBIT which can be more easily paralelized on a cluster computing environment.

Navigate to the audio folder to view the saved results, or view a short summary of top-scoring files by scrolling to the bottom of the notebook.

advanced use

for more advanced use of RIBBIT in larger workflows, consider using OpenSoundscape in Python.