Improved Automatic Bird Identification through Decision Tree based Feature Selection and Bagging

Mario Lasseck

Animal Sound Archive
Museum für Naturkunde Berlin
Mario.Lasseck@mfn-berlin.de

Abstract. This paper presents a machine learning technique for bird species identification at large scale. It automatically identifies about a thousand different species in a large number of audio recordings and provides the basis for the winning solution to the LifeCLEF 2015 Bird Identification Task. To process the very large amounts of audio data and to achieve similar good results compared to previous identification challenges new methods e.g. downsampling of spectrogram images for faster feature extraction, advanced feature selection via decision tree based feature ranking and bootstrap aggregating using averaging and blending were tested and evaluated.

Keywords: Bird Identification · Content Based Information Retrieval · Template Matching · Decision Tree based Feature Selection · Bagging · Blending

1 Introduction

Automatic identification of species from their sound can be a very useful computational tool for assessing biodiversity with many potential applications in ecology, bioacoustic monitoring and behavioral science [1]. Some examples of previous studies on species identification, especially of birds, are given in [2,3,4,5]. The approach towards automatic species identification presented here is a further development of ideas and methods already successfully applied in previous challenges. The NIPS4B 2013 Multi-label Bird Species Classification Challenge [6] hosted by Kaggle for example asked participants to identify 87 sound classes (songs, calls and instrumental sounds) of more than 60 different species in a large number of wildlife recordings. Last year the LifeClef2014 Bird Identification Task [7] challenged participants to identify 501 different species in almost 5000 audio recordings. This year the number of training and test files as well as the number of species to identify was increased once again. With almost a thousand species and over 33,000 audio files it is the largest computer-based bird identification challenge so far. A detailed description of the task, dataset and experimentation protocol can be found in [8], [20]. The task is among others part of the LifeCLEF 2015 evaluation campaign [9,10,11].

The features used for classification and methods to speed up feature extraction are introduced in section 2. In section 3 a two-pass training approach is proposed including feature ranking and selection. Bagging is used to improve classification results and different methods for aggregating model predictions are compared. Finally, submission results are evaluated in section 4 and briefly discussed in section 5.

2 Feature Engineering

There are two main categories of features used for classification: parametric acoustic features (see openSMILE Audio Statistics) and probabilities of species-specific spectrogram segments (see Segment-Probabilities). The feature sets are briefly described in the following sections. Similar features have been already successfully used in previous identification challenges and additional details can be found in [12,13,14].

2.1 openSMILE Audio Statistics

For each audio file a large number of acoustic features were extracted using the openSMILE Feature Extractor Tool [15]. The configuration file *emo_large.conf*, originally designed by Florian Eyben for emotion detection in human speech, was modified in several ways to better capture the characteristics of bird sounds. The changes relate primarily to the frame-wise calculated low-level descriptors (LLDs). For example the maximum frequency for pitch and Mel-spectrum was set to 11 kHz (instead of 500 Hz and 8 kHz). Also, the number of Mel Frequency Cepstral Coefficients (MFCC) was increased as well as the number of frequency bands for energy calculations. Furthermore, pitch- and spectral-related LLDs were added e.g. harmonics-to-noise ratio, raw F0, spectral skewness, kurtosis, entropy, variance and slope.

The all in all 73 LLDs consist of:

- 1 time domain signal feature
  - zero crossing rate (ZCR)
- 39 spectral features
  - Mel-spectrum bins 0-25
  - 25%, 50%, 75% and 90% spectral roll-off points
  - centroid, flux, entropy, variance, skewness, kurtosis and slope
  - relative position of spectral minimum and maximum
- 17 cepstral features
  - MFCC 0-16
- 6 pitch-related features
  - F0 (fundamental frequency, pitch)
  - voicing probability (degree of harmonicity)
  - F0raw (raw F0 candidate without thresholding in unvoiced/noisy segments)

  - HNR (log harmonics-to-noise ratio computed from the ACF)
  - F0env (F0 envelope with exponential decay smoothing)
  - voice quality (fundamental frequency ‘quality’ (= ZCR of ACF))
- 10 energy features
  - log energy
  - energy in frequency bands: 150-500 Hz, 400-1000 Hz, 800-1500 Hz, 1000-2000 Hz, 1500-4000 Hz, 3000-6000 Hz, 5000-8000 Hz, 7000-10000 Hz and 9000-11000 Hz

LLDs are calculated per audio frame (FrameSize 25 ms, StepSize 10 ms). To describe an entire audio recording, delta (velocity) and delta-delta (acceleration) coefficients were added to each LLD and finally 39 statistical functionals e.g. means, extremes, moments, percentiles and linear as well as quadratic regression were applied after smoothing all 219 feature trajectories via moving average (window length = 3). All in all, this sums up to 8541 (73×3×39) features per audio file. Further details regarding openSMILE and the acoustic features extracted can be found in the openSMILE book (http://www.audeering.com/research/opensmile) and the *OpenSmileForBirds_v2.conf* configuration file (http://www.animalsoundarchive.org/RefSys/LifeCLEF2015).

2.2 Segment-Probabilities

The second source used for features are Segment-Probabilities (*SegProbs*). For each species a set of representative segments also referred to as region of interests (ROIs) or templates was extracted from spectrogram images representing the acoustic content of audio files. These segments were then used to calculate Segment-Probabilities for each target file by finding the maxima of the normalized cross-correlation [16] between all segments and the target spectrogram image via template matching. A more detailed description regarding preprocessing, segmentation of spectrogram images and extraction of Segment-Probabilities can be found in [12] and [14].

Due to the very large amount of audio data, not all files belonging to a certain species were used as a source for segmentation. In a first session only short, good quality files (metadata: Quality = 1) without background species were selected for segment extraction. If the number of segments was smaller than a given threshold another file belonging to the same species was selected and so on. To ensure diversity and to capture the entire sound repertoire of a given species each file was chosen to belong to a different bird individual. To keep track of individuals an Individual-ID was assigned to each audio file. Two audio files of the same species were assigned the same Individual-ID if, according to the metadata, they were recorded by the same author on the same day. Individual-IDs were also used to accomplish a somewhat individual-independent species classification by creating "individual-independent" folds for cross-validation during training.

In the first session 262,232 segments were extracted from 2027 audio files of the training set with an average of 262 segments and 2 files (individuals) per species.

Fast Template Matching through prior Downsampling. When starting the template matching to collect Segment-Probabilities as described in [12] it quickly became apparent that sliding 262,232 templates over the spectrogram representation of all audio recordings was too time consuming (33,203 files in total). Even with modifications described in [14] it would have taken too long. Both methods apply a Gaussian blur on segments and target image before the actual template matching. This smoothing is a form of low-pass filtering to reduce detail. Interestingly, Gaussian smoothing is also used when reducing the size of an image. Before downsampling an image, it is common to apply a low-pass filter to ensure that spurious high-frequency information does not appear in the resampled image (aliasing). So if high-frequency information is discarded anyway why not apply a Gaussian blur and then downsample both template and target image by a factor 2 prior to the template matching? Together with other speed-related optimizations introduced in [14] (e.g. short-time Fourier transform (STFT) with only 50% overlap, restricting the template matching to a few pixels above and below the original vertical segment position along the frequency axes) this preprocessing reduces calculation time significantly while maintaining comparable results in finding maxima of segments within spectrograms. Proportions of spectrogram images and effects of low-pass filtering are visualized in Fig. 1 for an audio file of about 8 seconds taken from the training set (MediaId: 83).

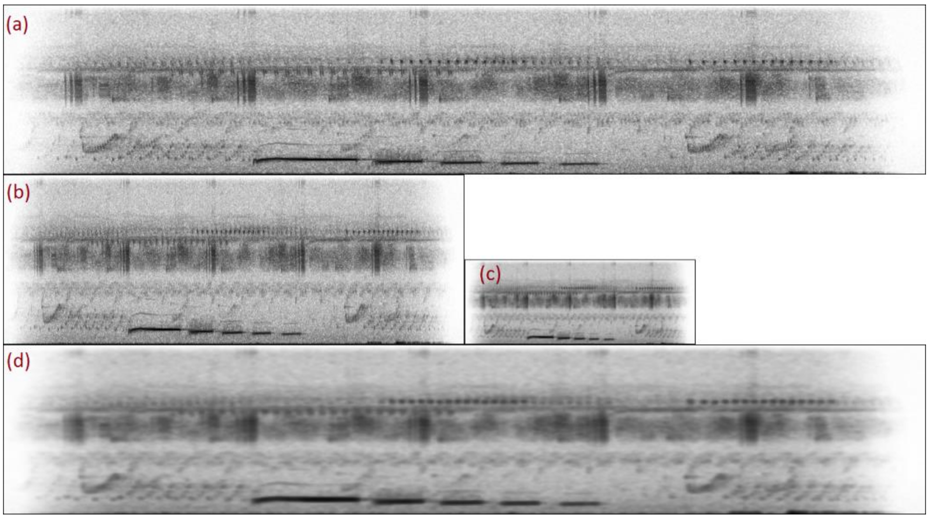

Fig. 1. (a) Original spectrogram image: STFT with 75% overlap, **(b)** STFT with 50% overlap, **(c)** Image for template matching: Downsampling of (b) by factor 2, **(d)** Loss of information: Expansion of (c) to the size of the original spectrogram image

3 Training, Feature Selection and Classification

Like in previous challenges classification was split up into several independent classification tasks by training one classifier per species (999 classes) following the binary relevance approach. For each species the classification problem was formulated as a multi-label regression task with the target function set to 1.0 for dominant species and 0.5 for all background species. For classification the scikit-learn library [17] was used (ExtraTreesRegressor) by training ensembles of randomized decision trees [18] with probabilistic outputs. Hyperparameter grid search was performed to improve classification results. For each classifier the following parameters and variations of tree-specific hyperparameters were used during training:

- number of "individual-independent" folds for cross-validation: 10
- number of estimators (trees in the forest): 500
- number of features to consider when looking for the best split: 10, 30
- minimum number of samples required to split an internal node: 5, 10

Best hyperparameters were chosen separately for each species by evaluating the Area Under the Curve (AUC) on predictions of held-out training files. To have a more realistic estimation and to improve generalization, Individual-IDs were used to create "individual-independent" folds for cross-validation. This way, recordings of the same bird were either part of the training or the validation set but not both. For the best runs the probability of occurrence for that species was predicted in all test files and averaged during cross-validation.

Decision Tree based Feature Ranking and Selection. For both feature sets (*SegProbs1* & *openSMILE*) training was performed in two passes. During the first pass feature importances returned by the classifier were cumulated for each species during hyperparameter variation and saved for later feature ranking. The importance of a feature was computed as the total reduction of the mean squared error brought by that feature. During the second pass classifiers were trained again, but this time with only a limited number of features, starting with the most important ones. To determine the optimal number of *SegProbs1* and *openSMILE* features for each species features were added in decreasing order of importance. For *SegProbs1* the number of selected features considered for training was 10, 50, 100, 150 and 500 and for *openSMILE* 50, 100, 150, 500, 3000 and 8541 (no feature selection). In Fig. 2 the frequency distribution regarding the optimal number of selected features (= best AUC per species) is given as bar chart for both feature sets.

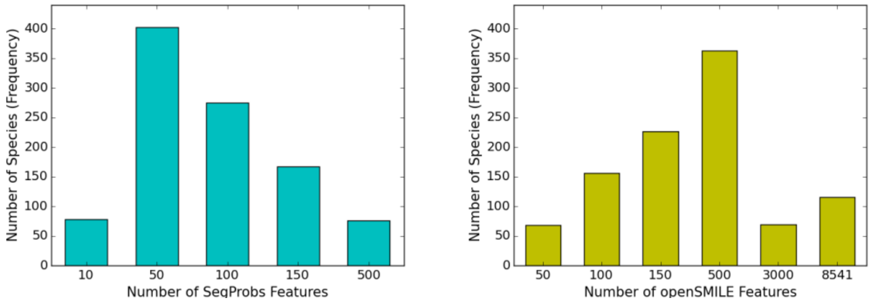

Fig. 2. Absolute frequency distribution regarding the optimal number of selected features for **left:** Segment-Probabilities and **right:** openSMILE feature set

For Segment-Probabilities this means using the 50 most important features only is most likely better than using the most important 100, 150 or 500 features to identify a species. For *openSMILE* using 500 important features on average is better than using 3000 or all of them. Figure 3 gives an impression to what extent the above described feature selection method improves classification results over the entire training set. Each column in the boxplot summarizes the best possible cross-validated AUC score achieved for each species during hyperparameter grid search and the dotted line shows the improvement regarding the mean AUC.

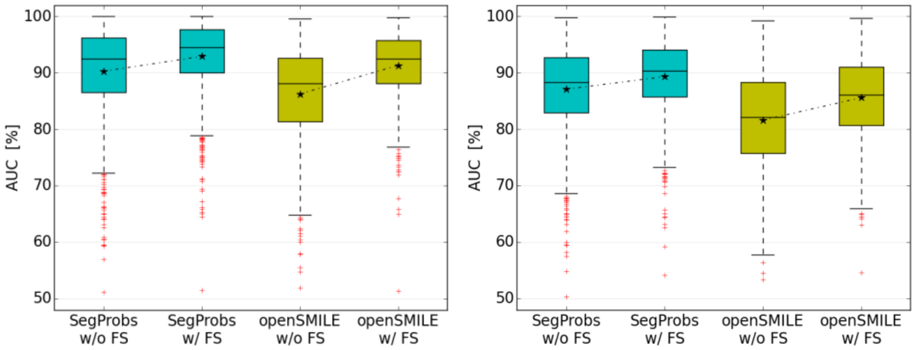

Fig. 3. AUC statistics: without (w/o FS) vs. with (w/ FS) feature selection for *SegProbs1* and *openSMILE* features, **left**: without background species, **right**: with background species

Bootstrap Aggregating. To further improve classification results additional feature sets were created by calculating further Segment-Probabilities for smaller parts of the training set (*SegProbs2* & *SegProbs3*). For this pupose additional segments were extracted from files belonging to species that did not exeeded an AUC score of 98% during cross-validation without background species in the previous training steps.

Again, only files belonging to individuals not processed before were chosen for further segmention. Segment-Probabilities calculated for last year's challenge were also reused as features (*SegProbsOld*). Properties of the different sets are summarized in Table 1.

Table 1. Properties of Segment-Probabilities feature sets (total number of features, average number of features per species, number of species covered)

Feature set	# Features (Segments)	# Features per Species (avg)	# Species
SegProbs1	262,232	262	999
SegProbs2	224,852	557	404
SegProbs3	245,003	620	395
SegProbsOld	492,753	985	500

For all sets feature selection based on prior feature ranking was performed as described above. Finally, four data sets (*C1_All*, *C2_2014*, *C3_2015* and *C4_Old*) were created for each species by combining different feature sets. Those subsets can be interpreted as bootstrap data sets where rows represent a subsampling of the training files and columns a subsampling of the feature space. The number of training files, test files and species associated with each data set are listed in the table below.

Table 2. Number of audio files and number species used in different bootstrap data sets

	BirdCLEF2014			BirdCLEF2015		
Data set	#Train Files	#Test Files	# Species	#Train Files	#Test Files	# Species
C1_All	9596	4299	500	15011	4297	499
C2_2014	7919	4299	404	3956	4297	-
C3_2015	-	-	-	8604	4297	395
C4_Old	9596	4299	500	-	-	-

The bootstrap data sets were used to train somewhat independent predictive models. The predictions of those models were than combined which is also known as bootstrap aggregating or bagging. In bootstrap aggregating the subsets are usually randomly drawn but here the data sets were chosen from a pragmatic point of view with regards to BirdCLEF 2014 vs. 2015 data and extraction of additional features for species with classification results below a certain threshold. Besides faster training, this ensemble method helps to reduce variance and improves stability. For all bootstrap data sets different models were trained using different feature sets and feature set combinations:

- Data subset 1: *C1_All*
  - *openSMILE* without and with prior feature selection (w/o & w/ FS)
  - *SegProbs1* (w/o & w/ FS)

- Data subset 2: *C2_2014*
    - *SegProbs2* (w/o & w/ FS)
    - *SegProbs2+SegProbs1* (w/ FS)
    - *SegProbs2+SegProbs1+openSMILE* (w/ FS)
- Data subset 3: *C3_2015*
    - *SegProbs3* (w/o & w/ FS)
    - *SegProbs3+SegProbs1* (w/ FS)
    - *SegProbs3+SegProbs1+openSMILE* (w/ FS)
- Data subset 4: *C4_Old*
    - *SegProbsOld* (w/o & w/ FS)

Combining the Outputs of Multiple Classifiers via Averaging and Blending. To combine predictions of the different models two methods were tested: simple averaging and blending (also known as stacking or stacked generalization [19]). In case of blending the outputs of different classifiers were used as training data for another classifier to approximate the same target function. For this second level classifier an ordinary least squares linear regression model was trained to figure out the combining mechanism or weighting of the individual predictions.

Post-processing of Predictions for Submission. The predictions returned by the classifiers assign a score to each species within each audio file (probability of occurrence as real value). After blending some prediction values were not within the requested interval [0,1]. To deal with this, all negative values were clipped to zero. Additionally, all values greater 0.6 were replaced using a hyperbolic tangent function (tanh). By passing predictions to this non-linear transfer function, values were all kept below 1.0. This way ranking was preserved especially among the top ranks that are most important when evaluating the Mean Average Precision (MAP). In a final step predictions of species not part of last year's challenge were set to zero for all files marked with Year = 'BirdCLEF2014'. Figures 4 and 5 show the progress of classification results using simple averaging compared to blending for stepwise aggregating predictions from and within bootstrap data sets followed by post-processing. Results are presented via AUC statistics (summarized for all species as boxplots) and MAP statistics (evaluated over the entire training set) within the same figure.

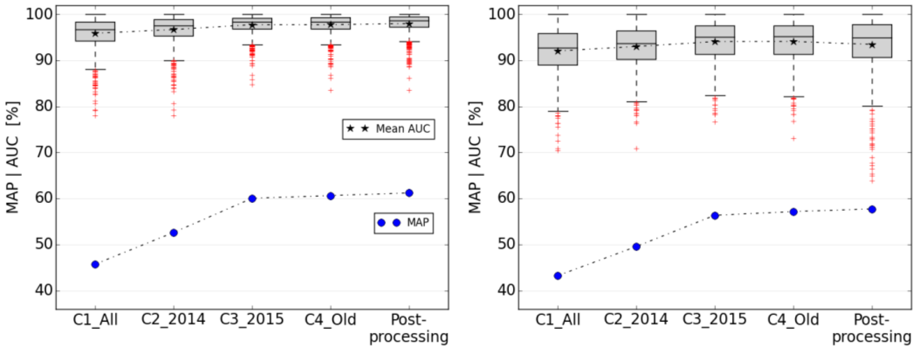

Fig. 4. Progress of classification results: Aggregating predictions via **Averaging**, **left**: w/o BS, **right**: w/ BS

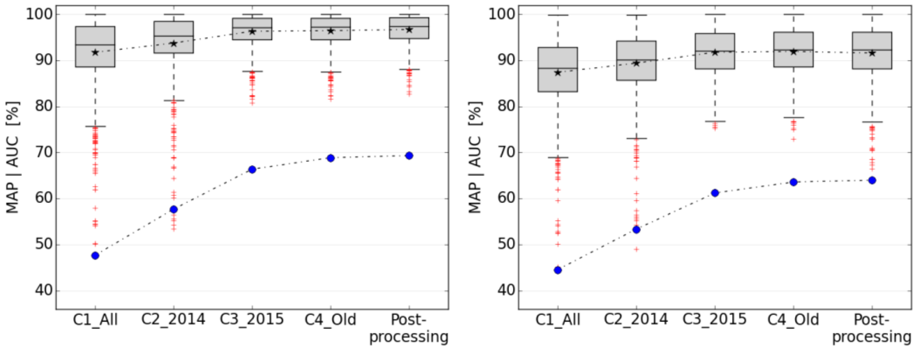

Fig. 5. Progress of classification results: Aggregating predictions via **Blending**, **left**: w/o BS, **right**: w/ BS

It is worth mentioning that AUC and MAP statistics are not perfectly correlated. Although averaging leads to better AUC statistics, blending yields much better MAP scores. Also, setting predictions of the 499 species new in 2015 to zero during post-processing for BirdCLEF2014 files increases MAP for both averaging and blending if background species are included for evaluation, whereas AUC statistics are getting worse in both cases (far more outliers below 1.5×IQR). This can be explained by the fact that those species are not among the dominant species in BirdCLEF2014 files but they do may appear as background species.

4 Submission Results

In Table 3 results of submitted runs are summarized using two evaluation statistics: mean of the Area Under the Curve (AUC) calculated per species and mean Average

Precision (MAP) on the public training and the private test set. All four runs outperformed the results of the other participating teams [20].

Table 3. Performance of submitted runs (without / with background species)

	Public Training Set		Private Test Set
Run	**Mean AUC [%]**	**MAP [%]**	**MAP [%]**
1	95.2 / 90.0	43.3 / 40.8	42.4 / 38.8
2	96.6 / 91.3	67.1 / 62.1	44.2 / 40.5
3	98.1 / 93.5	61.2 / 57.7	44.2 / 41.1
4	96.7 / 91.6	69.3 / 64.0	45.4 / 41.4

For the first run only *SegProbs1* and *openSMILE* features were used for classification. Different models were trained on the entire training set with and without prior feature selection. The predictions of the models were than combined via blending followed by post-processing as described above. For the second run additional classifiers were trained on smaller subsets of the training files (*C2_2014*, *C3_2015* and *C4_Old*). For each subset a distinct set of additional features (*SegProbs2*, *SegProbs3* and *SegProbs-Old*) was used for training. For this run features from one subset were not part of any other subset. Again, feature selection was performed individually for each species and each bootstrap data set. Predictions of all subset models including the ones from the first run were aggregated via blending and post-processed. For the third and fourth run additional models were trained for each bootstrap data set including also selected features from other subsets. The difference however between these two runs is that averaging was used for run three whereas blending was used for the final and best performing fourth run to aggregate model predictions. In Fig. 6 results for all submitted runs are visualized as combination of AUC boxplots and MAP scores.

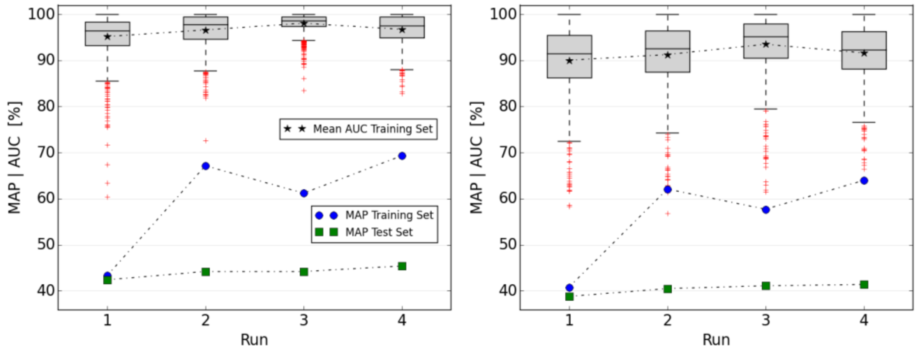

Fig. 6. AUC and MAP statistics of submitted runs, **left**: w/o BS, **right**: w/ BS

5 Discussion

Downsampling spectrogram images prior to template matching significantly reduces calculation time. Luckily, the here occurring loss of information can actually be considered as a feature, not a bug. Too much detail is rather disturbing and distracting when comparing call or song elements between different individuals of the same species. For some birds maybe even a downsampling factor greater than 2 produces equal or even better results? Faster template matching means – especially when dealing with so many species and such large amounts of audio data – being able to extract more segments and calculate more Segment-Probabilities. But using more features does not necessarily lead to better identification results. Reducing the number of features to the most important ones has several advantages. Less but meaningful features improve prediction accuracy, reduce over-fitting and therefore increase generalization power. Besides faster training and prediction, a smaller set of features also reduces, once the model is trained, prediction time of new and unseen audio material due to faster feature extraction. Especially in case of Segment-Probabilities from now on only relevant segments need to be considered for template matching.

Due to the large amount of data and limited training time the searching for the optimal number of features per species was not very extensive. Only a few manual selected candidates were evaluated. A finer grid search or a more sophisticated way to approach the true optimum, for example using a binary search algorithm, might have led to better results.

When looking at the different MAP scores for training and test files in Fig. 6 it becomes clear that most of the progress achieved by bagging on the training set is due to over-fitting. This could be partly explained by the fact that only for the first dataset *C1_All* an "individual-independent" training approach with accordingly selected folds was used whereas for all other subsets common stratified folds were used for cross-validation. Another reason might be that all bootstrap data sets used similar features and equal classification methods and therefore model predictions were not independent or uncorrelated enough to significantly boost classification results when combining them. Nevertheless bagging increased average prediction scores also for test files and could clearly improve submission results.

Acknowledgments. I would like to thank Hervé Glotin, Hervé Goëau, Andreas Rauber and Willem-Pier Vellinga for organizing this challenge, Alexis Joly and Henning Müller for coordination, the Xeno-Canto foundation for nature sounds for providing the audio data and the French projects Pl@ntNet (INRIA, CIRAD, Tela Botanica) and SABIOD Mastodons for supporting this task. I also want to thank Dr. Karl-Heinz Frommolt for supporting my work and providing me with access to resources of the Animal Sound Archive [21].

References

1. Frommolt K-H, Bardeli R, Clausen M (2008) ed. Computational bioacoustics for assessing biodiversity. Proc. of the int. expert meeting on IT-based detection of bioacoustical patterns
2. Bardeli R et al. (2009) Detecting bird sounds in a complex acoustic environment and application to bioacoustic monitoring. Pattern Recognition Letter: 31, 23, 1524–1534
3. Briggs F et al. (2012) Acoustic classification of multiple simultaneous bird species: A multi-instance multi-label approach. The Journal of the Acoustical Society of America 131: 4640.
4. Potamitis I (2014) Automatic Classification of Taxon-Rich Community Recorded in the Wild. PLoS ONE 9(5): e96936. doi: 10.1371/journal.pone.0096936
5. Stowell D, Plumbley MD (2014) Automatic large-scale classification of bird sounds is strongly improved by unsupervised feature learning. PeerJ 2:e488
6. https://www.kaggle.com/c/multilabel-bird-species-classification-nips2013
7. http://www.imageclef.org/2014/lifeclef/bird
8. Goëau H, Glotin H, Vellinga WP, Rauber A (2015) LifeCLEF Bird Identification Task 2015, In: CLEF working notes 2015
9. Joly A, Müller H, Goëau H, Glotin H et al. (2015) LifeCLEF 2015: multimedia life species identification challenges, In: Proceedings of CLEF 2015
10. Cappellato L, Ferro N, Jones G, San Juan E, eds. (2015). CLEF 2015 Labs and Workshops, Notebook Papers. CEUR Workshop Proceedings (CEUR-WS.org), ISSN 1613-0073, http://ceur-ws.org/ Vol-1391/.
11. Mothe J, Savoy J, Kamps J et al. (ed.), Experimental IR meets Multilinguality, Multimodality, and Interaction. Sixth International Conference of the CLEF Association, CLEF'15, Toulouse, September 8-11, 2015. Proceedings. LNCS, vol. 9283. Springer, Heidelberg (2015)
12. Lasseck M (2013) Bird Song Classification in Field Recordings: Winning Solution for NIPS4B 2013 Competition, In: Glotin H. et al. (eds.). Proc. of int. symp. Neural Information Scaled for Bioacoustics, sabiod.org/nips4b, joint to NIPS, Nevada, dec. 2013: 176-181
13. Lasseck M (2014) Large-scale identification of birds in audio recordings. In: Working notes of CLEF 2014 conference
14. Lasseck M (2015) Towards Automatic Large-Scale Identification of Birds in Audio Recordings, In: Proceedings of CLEF 2015
15. Eyben F, Weninger F, Gross F, Schuller B (2013) Recent Developments in openSMILE, the Munich Open-Source Multimedia Feature Extractor, In: Proc. ACM Multimedia (MM), Barcelona, Spain, ACM, ISBN 978-1-4503-2404-5, pp. 835-838, October 2013, doi:10.1145/2502081.2502224
16. Lewis JP (1995) Fast Normalized Cross-Correlation, Industrial Light and Magic
17. Pedregosa F et al. (2011) Scikit-learn: Machine learning in Python. JMLR 12, pp. 2825-2830
18. Geurts P et al. (2006) Extremely randomized trees, Machine Learning, 63(1), 3-42
19. Wolpert DH (1992) Stacked generalization. Neural Networks, 5:241–259
20. http://www.imageclef.org/lifeclef/2015/bird
21. Animal Sound Archive, http://www.animalsoundarchive.org