2013 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING, SEPT. 22–25, 2013, SOUTHAMPTON, UK

THE 9TH ANNUAL MLSP COMPETITION: NEW METHODS FOR ACOUSTIC CLASSIFICATION OF MULTIPLE SIMULTANEOUS BIRD SPECIES IN A NOISY ENVIRONMENT

Forrest Briggs[1,2], *Yonghong Huang*[1], *Raviv Raich*[1,2],
Konstantinos Eftaxias[1], *Zhong Lei*[1], *William Cukierski*[1],

Sarah Frey Hadley[2], *Adam Hadley*[2], *Matthew Betts*[2], *Xiaoli Z. Fern*[2], *Jed Irvine*[2], *Lawrence Neal*[2],

Anil Thomas[3], *Gábor Fodor*[3], *Grigorios Tsoumakas*[3], *Hong Wei Ng*[3], *Thi Ngoc Tho Nguyen*[3],
Heikki Huttunen[3], *Pekka Ruusuvuori*[3], *Tapio Manninen*[3], *Aleksandr Diment*[3], *Tuomas Virtanen*[3],
Julien Marzat[3], *Joseph Defretin*[3], *Dave Callender*[3], *Chris Hurlburt*[3], *Ken Larrey*[3], *Maxim Milakov*[3]

1 – Competition Organizer, 2 – Data Collection, Labeling & Curation, 3 – Competition Entrant

1. INTRODUCTION

Birds have been widely used as biological indicators for ecological research. They respond quickly to environmental changes and can be used to infer about other organisms (e.g., insects they feed on). Traditional methods for collecting data about birds involves costly human effort. A promising alternative is acoustic monitoring. There are many advantages to recording audio of birds compared to human surveys, including increased temporal and spatial resolution and extent, applicability in remote sites, reduced observer bias, and potentially lower cost. However, it is an open problem for signal processing and machine learning to reliably identify bird sounds in real-world audio data collected in an acoustic monitoring scenario. Some of the major challenges include multiple simultaneously vocalizing birds, other sources of non-bird sound (e.g., buzzing insects), and background noise like wind, rain, and motor vehicles.

The 9th annual MLSP competition presented a real-world dataset of bird sounds collected in field conditions. The goal of the challenge was to do develop a classifier which predicts the set of bird species present in a given ten-second audio recording. The competition was hosted on Kaggle.com, a platform for data mining competitions. Participation in this competition was quite extensive; 79 teams participated, and 8 out of the 10 top-ranking teams submitted a two-page summary of their proposed methods. This paper summarizes the results of the competition, and highlights the ideas from those summaries.

2. DATASET

The audio dataset for this challenge was collected in the H. J. Andrews (HJA) Long-Term Experimental Research Forest, in the Cascade mountain range of Oregon. Since 2009, members of the Oregon State University Bioacoustics group have collected over 10TB of audio data in HJA using Songmeter audio recording devices. A Songmeter has two omnidirectional microphones, and records audio in WAV format to flash memory. A Songmeter can be left in the field for several weeks at a time before either its batteries run out, or its memory is full.

HJA has been the site of decades of experiments and data collection in ecology, geology and meteorology. This means, for example, that given an audio recording from a particular day and location in HJA, it is possible to look up the weather, vegetative composition, elevation, and much more. Such data enables unique discoveries through cross-examination, and long-term analysis.

Previous experiments on supervised classification using multi-instance and/or multi-label formulations have used audio data collected with song meters in HJA [5, 3, 4, 17, 19]. The dataset for this competition is similar to, but perhaps more difficult than that dataset used in these prior works; in earlier work care was taken to avoid recordings with rain and loud wind, or no birds at all, and all of the recordings came from a single day.

In this competition, we consider a new dataset which includes rain, wind, and no-bird recordings, and is a representative sample of HJA in 2009 and 2010 at 13 sites (Fig. 2). The full dataset consists of 645 ten-second audio recordings in uncompressed WAV format (16kHz sampling frequency, 16 bits per sample, mono). There are 19 species of bird in the dataset (Table 1). The subset of 645 recordings chosen for

978-1-4799-1180-6/13/$31.00 ©2013 IEEE

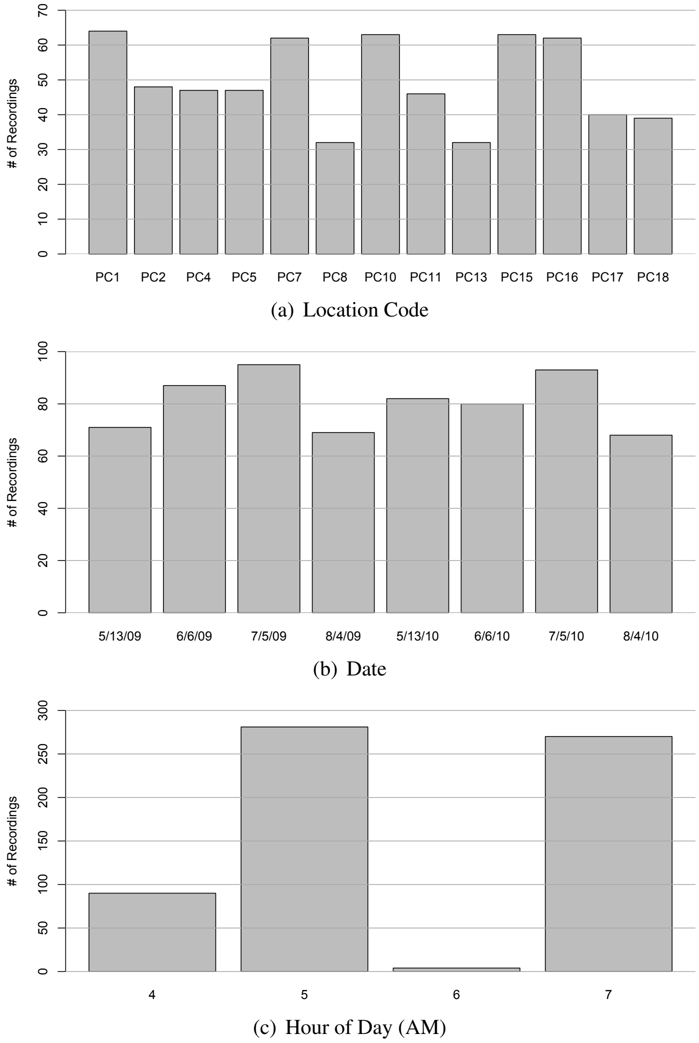
Table 1. The 19 bird species in the dataset.

Code	Name
BRCR	Brown Creeper
PAWR	Pacific Wren
PSFL	Pacific-slope Flycatcher
RBNU	Red-breasted Nuthatch
DEJU	Dark-eyed Junco
OSFL	Olive-sided Flycatcher
HETH	Hermit Thrush
CBCH	Chestnut-backed Chickadee
VATH	Varied Thrush
HEWA	Hermit Warbler
SWTH	Swainson's Thrush
HAFL	Hammond's Flycatcher
WETA	Western Tanager
BHGB	Black-headed Grosbeak
GCKI	Golden Crowned Kinglet
WAVI	Warbling Vireo
MGWA	MacGillivray's Warbler
STJA	Stellar's Jay
CONI	Common Nighthawk

this competition are intended to provide a good coverage of all recording sites (Fig. 1a), several days over multiple years (Fig. 1b), and several hours of day around dawn, when birds are most active (Fig. 1c).

Each ten-second audio recording is paired with a set of species that are present. These label sets were obtained by listening to the audio and looking at spectrograms. Several experts inspected each recording, and each provided their own label set, along with estimates of their confidence. The final label set was formed by confidence-weighted majority voting.

The WAV filenames encode the location (one of the 13 sites), and the date/time that the recording was collected. Participants were allowed to use the location information in their classifiers (indeed, this proved very useful), but we prohibited use of the date/time information.



(a) Location Code

(b) Date

(c) Hour of Day (AM)

Fig. 1. Recordings in the dataset, counted in different ways.

3. PROBLEM STATEMENT & EVALUATION

This challenge was formulated as a multi-label classification problem. Formally, each recording in the dataset R_i is paired with a set of species $Y_i \subset \{1, \ldots, c\}$ where $c = 19$. The task is to predict the probability that each species is present, given a recording, i.e. $P(j \in Y_i | R_i)$ for $j = 1, \ldots, c$. Classifiers are evaluated based on the "micro" area under the receiver operating characteristic curve (AUC) [15, 5].

The dataset was split randomly into 50% training set and 50% test set. Furthermore, the test set was divided into 1/3 "public test" and 2/3 "private test." During the competition, only the training set labels were provided. Participants were able to submit predictions on the public test set twice per day, and immediately receive a micro-AUC score. However, the final ranking for the competition was determined by micro-AUC on the private test set, which was not available to participants until after the competition ended. This methodology prevented participants from using their daily submission quota to reverse-engineer the private test set labels.

4. BASELINE METHOD

Participants were free to develop methods completely from scratch, using only the raw WAV audio files as input. However, there are many steps to go from audio to a predicted set of species, so to reduce barriers to participation, we provided a baseline method from prior work [5], which produces intermediate representations of the data through a sequence of steps. Many teams used some components of the baseline method. The steps in the baseline method, and data we pro-

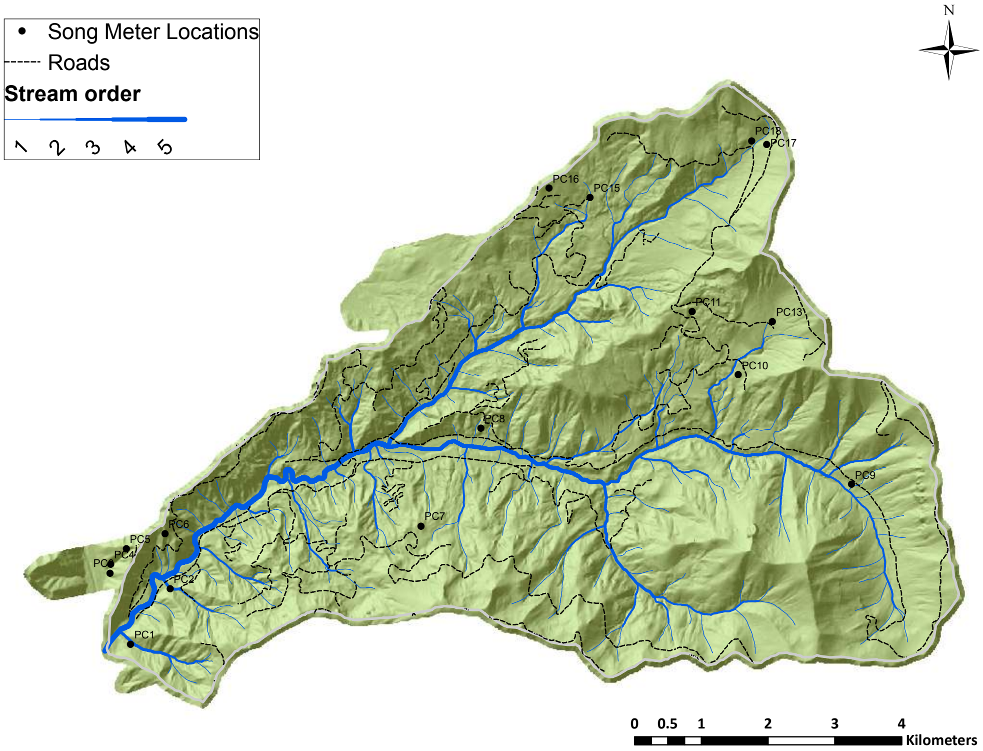


Fig. 2. Song meter data collection locations in the H. J. Andrews Experimental Forest.

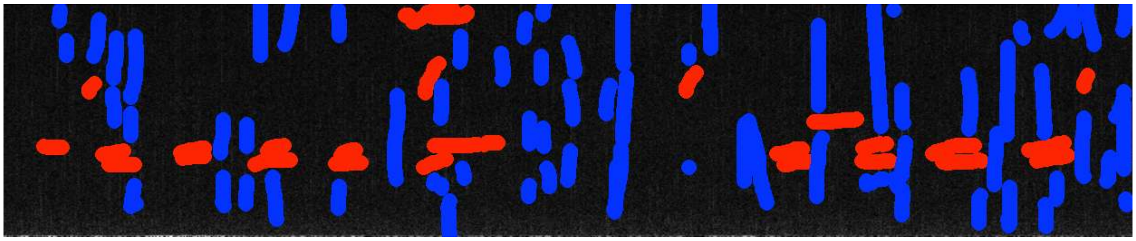


Fig. 3. Manually labeled spectrogram with example of correct segmentation. Red = bird, blue = rain. This spectrogram corresponds to 10 seconds of audio.

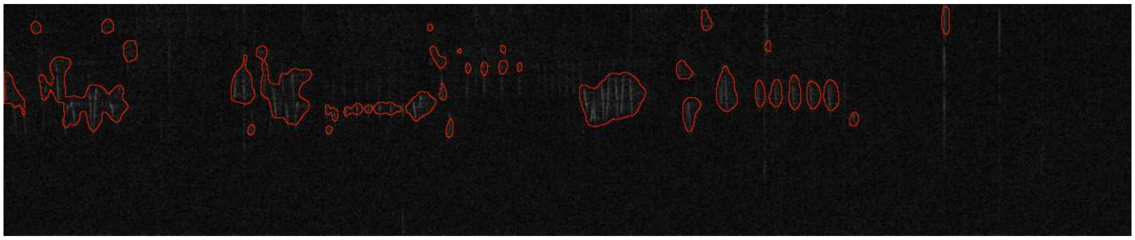


Fig. 4. Automatic segmentation of a spectrogram corresponding to 10-seconds of audio. This recording contains both rain and bird sound. In some cases the segmentation successfully ignored the rain and isolated bird sound, but in others in mistakenly labeled some rain as bird sound.

vided are:

Spectrograms – The raw audio signal is covered into a spectrogram (an image representing the sound), by dividing it into frames, and applying the FFT to each frame. We provided BMP images of the spectrograms.

Noise reduction – The frequency profile of stationary noise (such as wind and streams) is estimated from low energy frames, then the spectrogram is attenuated to suppress the background noise while preserving bird sound [5]. We provide a second set of noise-reduced spectrograms in BMP format. Note this stage of processing does not address rain.

Segmentation – Each spectrogram is divided into a collection of regions using a supervised time-frequency segmentation algorithm [5]. We manually annotated 20 spectrograms as examples of correct segmentation, by drawing over the areas corresponding to bird sound in red, and rain-drop in blue (Fig. 3). Because there are a large number of pixels in each spectrogram, we subsampled 30% of red pixels as positive examples, 30% of blue pixels as negative examples, and 4% of uncolored pixels as negative examples. From the 20 annotated spectrograms, this sampling process yields 467,958 examples. Each pixel is described by a feature vector with the following elements:

- The raw pixel intensity of all pixels in a 17 × 17 box around the pixel (this gives a 17^2 = 289-d feature).
- The average intensity of all pixels in that box (1-d).
- The y-coordinate of the pixel, which corresponds to frequency (1-d).
- The raw pixel intensity of all pixels in the same column as the pixel (256-d) (this feature was not present in the original work [5]; it was introduced for this competition to help the classifier differentiate between patterns characteristic of rain, and bird sound).

A Random Forest [2] classifier is trained on the positive and negative examples[1]. Then the trained Random Forest classifier is applied to each pixel in every spectrogram, which gives a probability for the pixel to be bird sound. The probabilities may be noisy when viewing individual pixels in isolation, so they are averaged over a neighborhood by applying a Gaussian blur to an image of the probabilities, with a kernel parameter $\sigma = 3$. The blurred probabilities are then compared to a threshold of 0.4. Pixels with probabilities above the threshold are considered to be bird sound and pixels with probabilities below the threshold are considered background. Figure 4 shows an example of the output of this segmentation process.

Segment Features – Each segment is associated with a 38-d feature vector which characterizes its shape, texture, and noise-robust profile statistics [5]. The segment features can be used directly as part of a multi-instance multi-label formulation, as in [5, 3, 4, 17], however most participants did not take this approach.

Histogram of Segments – To map the problem directly into a multi-label classification formulation, it is necessary to have a single feature vector summarizing each recording, rather than a collection of feature vectors summarizing the segments in the recording. For this purpose, we provided a "histogram of segments" (HOS) feature. The segments were clustered using k-means++ [1] to form a codebook, then each recording was represented by counting the number of times each segment it contains is closest to each cluster center [6].

Using the HOS features, we applied binary relevance (BR) with Random Forest (RF) as the base classifier (this method was used for classification of data from HJA, and bird sounds collected with a mobile phone in [6]). This baseline method (designated BR + RF + HOS) achieved an AUC of .85576 and a rank of 46/81 (Table 2).

5. RESULTS & NEW METHODS

Table 2 shows the AUC results achieved by all teams, and Figure 5 shows a histogram of AUC values. This section highlights key ideas from new methods proposed by top-ranking entrants in the competition.

5.1. Segmentation

Many of the top 10 teams used the baseline segmentation algorithm. However, a few teams devised their own segmentation method, or modified the baseline method. In particular, team *beluga* (rank 1) proposed a new segmentation algorithm consisting of the following steps:

- Gaussian smoothing
- Thresholding applied to the intensity gradient of the smoothed image (this step roughly produces outlines of the segments)
- Fill holes
- Remove small segments

Team *Herbal Candy* (rank 2) proposed a "sub-band energy ratio method" for segmentation with the following steps:

- Pre-emphasis, bandpass filtering, and a median filter to remove salt and peper noise
- Compute a sub-band spectrogram with a lower frequency resolution (16 bands)
- Apply an energy threshold to the sub-band spectrogram to obtain segments
- Scale the segmentation back up to the original spectrogram size

[1] The segmentation Random Forest parameters are: 100 trees, maximum depth of 10, histograms are stored in leaves

Team *JM-JD* (rank 10) applied a similar method to the baseline segmentation algorithm, but tuned the intensity threshold by cross-validation, and used a Canny edge detector [7], to extract chained edge pixel lists for the perimeter of each segment, and discarded segments with a perimeter of less than 20 pixels (this is different from how small segments were discarded in the baseline method).

Many of the teams computed features based on either the baseline segmentation or one of the above segmentation methods. Some of the teams also constructed features in ways that did not involve segmentation of the spectrogram into regions.

5.2. Features

Most of the top 10 ranked teams formulated the task as a multi-label classification problem, hence their methods involved constructing a fixed-length feature vector to describe each 10-second spectrogram. Some of the features used describe segments, frames, or patches, while others characterize the recording as a whole. Features characterizing parts of the spectrum are typically summarized in some way to produce a spectrogram-level feature.

Herbal Candy computed a suite of features to describe each segment including a subset of the baseline features (minimum/maximum frequency, bandwidth, duration, area, perimeter, non-compactness, and rectangularity), as well as frame-averaged mel-frequency cepstral coefficients (MFCC), delta-MFCC, linear prediction cepstral coefficients (LPCC), and spectral properties (sub-band energy, sub-band entropy, centroid, roll-off and flux).

Bag-of-words, histogram, and dictionary representations were widely used. The baseline histogram of segments features were used by teams *beluga*, *Anil Thomas*, *default*, *Tap & Huttunen*, and *windmills*. Furthermore, *Anil Thomas* also constructed a second histogram of segments with $k = 10$ clusters. *Herbal Candy* constructed two bag-of-words feature sets, one using normalized counts as in the HOS features, and another using the distance to the nearest codeword rather than the count, following [20]. *JM–JD* also constructed a bag-of-words representation, but rather than a histogram, they first clustered segment features, then constructed a feature vector consisting of the average Hausdorff distance [23] from the set of segments in a spectrogram to the set of segments in each cluster. *Tap & Huttenun* used a more sophisticated dictionary learning approach [14], implemented in the SPAMS toolbox [18]. Their approach was to represent rectangular patches of the spectrogram as a sparse linear combination of dictionary atoms. The dictionary atoms were obtained by minimizing the L_1-regularized Euclidean distance between each original patch and its optimal linear reconstruction from atoms.

A simple summary feature used by team *default* was the mean and standard deviation of the height, width, and area of the bounding boxes for each segment found by the baseline segmentation method.

Another method of summarizing the patterns present in a spectrogram as a fixed-dimensional feature vector is to use template matching. In particular, a set of template patterns (segments) are chosen, then a feature vector is constructed by computing the maximum normalized cross-correlation of each template with the spectrogram. This approach was used by *beluga* and *windmills*.

Some spectrum-summarizing features divided the spectrogram into frequency bands, then summarized each bands with statistics. In particular, *Anil Thomas* divided the spectrogram into 16 bands, then to avoid contribution from background noise, computed the mean intensity of pixels above a threshold. Team *default* used a 168-dimensional statistical spectrogram descriptor (SSD) feature extracted using the rp_extract software [16], which consists of the mean, variance, skewness, kurtosis, min, max, and median statistic within each of 24 bark bands.

It was widely reported that use of location information improved classification accuracy. Many teams encoded the location where the data was collected a single categorical feature (*beluga*, *Anil Thomas*, *default*, and *JM–JD*). Alternatively, some teams computed the empirical probability of each species to occur at each site from the training data, and formed a 19 dimensional vector (corresponding to 19 species) from these probabilities, then used the probability vector corresponding to the site from which each recording came as part of the feature vector for that recording (*Anil Thomas*, and *Herbal Candy*). Team *JM–JD* also used approximate distance to stream as a feature, computed from the map of HJA provided with the competition.

Several teams included features which were designed to help differentiate between recordings consisting only of background noise and recordings containing bird sound. For example, *Anil Thomas* and *default* used the number of segments detected by the baseline segmentation method as a feature. *Herbal Candy* used entropy, and the correlation coefficient between the average noise frame and signal frame to characterize the amount of interesting sound in a clip.

Tap & Huttunen's final classifier consisted of an averaged ensemble of several different classifiers, each using its own feature set. However, some of these features included MFCCs, Local Binary Pattern features [8], and the vector of frequencies over time at which the maximum amplitude occurred.

5.3. Classifiers

Many of the teams focussed primarily on feature design, and adopted a relatively simple approach for classification. Binary relevance with Random Forest was the most popular method (used by *beluga*, *Anil Thomas*, *default*, *Tap & Huttunen*, and *windmills*). Team *default* used the Mulan software package for learning multi-label learning [21]. Extremely Randomized Trees [9] were used by *Herbal Candy*, and *Tap & Hut-*

tunen. *Tap & Huttunen* also used L_1-regularized logistic regression [11] and k-nearest neighbors as part of an ensemble of classifiers. *JM–JD* used the multi-label radial basis function (ML-RBF) algorithm [22].

Many of the recordings had no species present, in which case the optimal prediction is 0 probability for all species. Team *default* created two different models, and switched between them based on wether the baseline segmentation method detected zero or more segments. If zero segments were detected, then a model using only the SSD and location features were used. Otherwise, a model using those features as well as features summarizing the segments was used. Team *windmills* devised a two-stage classifier, where the first stage predicted the probability that any species was present in a recording, then the second stage modeled the conditional probability of each species given that some bird was present.

Maxim Milakov (author of the nnForge library[2]) proposed a method based on convolutional neural nets [12]. In this method, the raw unfiltered spectrograms are given to the network as input, and the output layer has one neuron for each species. Max-pooling was used, i.e. the probability of a bird singing within the given 10-second interval is the max of the probability of it singing in 32 smaller overlapping intervals. The network was trained using the stochastic diagonal Levenberg-Marquadt algorithm [13], with dropout regularization [10] applied to the last convolutional layer. To encourage time-shift invariance, each example was randomly shifted during training, and during testing the output was averaged over five random shifts of the input. Remarkably, this method achieved high accuracy (rank 4/81) without extensive feature engineering, or using the location information (which helped other methods significantly).

5.4. Conclusion

Results of this competition demonstrate that reliable automatic species recognition using machine learning is feasible with audio collected in real-world field conditions, with difficulties including multiple simultaneously vocalizing birds, and background noises such as rain. Decision tree ensembles were the most popular type of classifier used, and generally achieved high accuracy. More interesting variations were displayed in the feature design part of the problem. Based on results of the top-ranking teams, some of the most useful features are the histogram of segments provided in the baseline, template matching features, signal descriptors not based on segmentation, and location. Alternatively, convolutional neural nets achieved excellent results without extensive feature engineering, hence further investigation of such methods is warranted.

[2] `http://milakov.github.io/nnForge/`

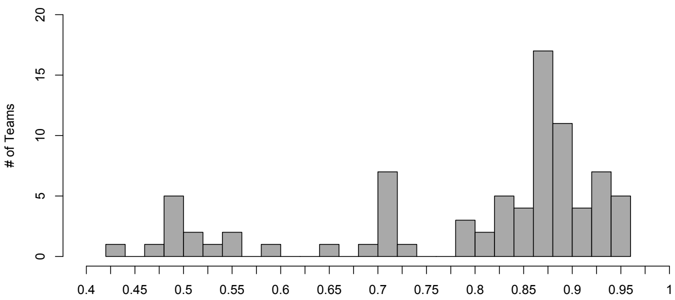


Fig. 5. Histogram of AUC scores achieved by teams.

6. ACKNOWLEDGEMENTS

Collection and preparation of this dataset was partially funded by NSF grant DGE 0333257, NSF-CDI grant 0941748, NSF grant 1055113, NSF grant CCF-1254218, and the College of Engineering, Oregon State University. We would also like to thank Iris Koski, Joshua Stagner, Evan Jackson, Lauren Smith, April Bartelt, Marcel Villar, and Sean Ashe for their contributions in data collection and labeling.

References

[1] David Arthur and Sergei Vassilvitskii. k-means++: the Advantages of Careful Seeding. In *SODA '07: Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007.

[2] L. Breiman. Random forests. *Machine learning*, 45(1): 5–32, 2001.

[3] F. Briggs, X.Z. Fern, and R. Raich. Rank-loss support instance machines for miml instance annotation. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 534–542. ACM, 2012.

[4] F. Briggs, X.Z. Fern, R. Raich, and Q. Lou. Instance annotation for multi-instance multi-label learning. *Transactions on Knowledge Discovery from Data (TKDD), 2012*, 2012.

[5] F. Briggs, B. Lakshminarayanan, L. Neal, X.Z. Fern, R. Raich, S.J.K. Hadley, A.S. Hadley, and M.G. Betts. Acoustic classification of multiple simultaneous bird species: A multi-instance multi-label approach. *The Journal of the Acoustical Society of America*, 131:4640, 2012.

[6] F. Briggs, X. Z. Fern, and J. Irvine. Multi-Label Classifier Chains for Bird Sound. *ICML 2013 Workshop on Machine Learning for Bioacoustics*, 2013.

Table 2. Rank, team, and AUC.

Rank	Team	AUC	Rank	Team	AUC
1	beluga[a]	.95612	42	Jonathan Simon	.86381
2	Herbal Candy[b]	.9505	43	Phoenix	.86328
3	Anil Thomas[c]	.94488	44	Parthiban Gowthaman	.86121
4	Maxim Milakov[d]	.94135	45	Bionic Insight	.85674
5	wweight	.94061	46	BR + RF + HOS*	.85576
6	Windmills[e]	.93764	47	desperate data miners	.84582
7	Tap & Heikki Huttunen[f]	.93759	48	Friendly snake	.84508
8	default[g]	.93527	49	BayesianAveraging	.83592
9	doubleshot	.93014	50	SABIOD team	.82843
10	JM–JD[h]	.92544	51	marger	.82726
11	Birdooma & jajo	.92389	52	Natarajan Sundaram	.82676
12	Matt Sco	.92238	53	IAN	.82625
13	AMPires	.91985	54	saket kunwar	.81126
14	MMDL	.91559	55	TimGarnsey	.80484
15	Luxtorpeda	.91461	56	Jacob M	.797
16	megasoft	.90038	57	Thakur Raj Anand	.79136
17	Team Rocket	.89944	58	Revenge of the Dodo	.78668
18	FY-AR	.89899	59	PRLab	.73056
19	Jeremy Benthams	.89804	60	Peter	.7167
20	developerX	.89296	61	rafonseca	.71479
21	DSPCom	.8879	62	AR2	.71337
22	TeamJon	.88438	63	bln	.71228
23	T	.88413	64	AZERLIA	.70799
24	area	.88321	65	charwizard	.7058
25	No bird	.8831	66	utdiscant	.70026
26	saraswathi	.88096	67	JDai	.69241
27	:-)	.8804	68	mn1aC	.64471
28	ikretus	.87988	69	mr1yh1	.58314
29	Bojan Vujatovic	.87628	70	CityUniMIRG	.5597
30	Pajaros	.87599	71	shark8me	.5465
31	Khagesh Patel	.87451	72	Krzysztof Babinski	.53519
32	zeon	.87372	73	JACR	.51347
33	fixalytics–solutions	.87161	74	Afroz Hussain	.5
34	Ankush Shah	.87099	75	sample–submission.csv*	.5
35	TeamGR	.87083	76	Domcastro	.5
36	Seppo Fagerlund	.87022	77	Wendy Kan	.5
37	bmax	.87009	78	Horia	.5
38	Mariko	.86977	79	paper plates	.5
39	Owen	.86942	80	Sven Cornelis	.47508
40	Abhishek & Issam	.86672	81	Iain Rice	.42235
41	simonmab	.86573			

* – Baseline method

Team	Authors
(a)	Gábor Fodor
(b)	Hong Wei Ng, Thi Ngoc Tho Nguyen
(c)	Anil Thomas
(d)	Maxim Milakov
(e)	Dave Callender, Chris Hurlburt, Ken Larrey
(f)	Heikki Huttunen, Pekka Ruusuvuori, Tapio Manninen, Aleksandr Diment, Tuomas Virtanen
(g)	Grigorios Tsoumakas
(h)	Julien Marzat, Joseph Defretin

[7] John Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):679–698, 1986.

[8] Yandre MG Costa, LS Oliveira, Alessandro L Koerich, Fabien Gouyon, and JG Martins. Music genre classification using LBP textural features. *Signal Proc.*, 92: 2723–2737, 2012.

[9] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Mach. Learn.*, 63(1):3–42, April 2006. ISSN 0885-6125. doi: 10.1007/s10994-006-6226-1.

[10] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[11] Heikki Huttunen, Tapio Manninen, and Jussi Tohka. Bayesian error estimation and model selection in sparse logistic regression. In *Proceeding of IEEE Machine Learning for Signal Processing Workshop (MLSP)*, pages 801–808, Sept. 2013.

[12] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[13] Yann LeCun, Léon Bottou, Genevieve B Orr, and Klaus-Robert Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–50. Springer, 1998.

[14] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Ng. Efficient sparse coding algorithms. In *Advances in neural information processing systems*, pages 801–808, 2006.

[15] David D. Lewis. Evaluating text categorization. In *In Proceedings of Speech and Natural Language Workshop*, pages 312–318. Morgan Kaufmann, 1991.

[16] Thomas Lidy and Andreas Rauber. Evaluation of feature extractors and psycho-acoustic transformations for music genre classification. In *Proceedings of the Sixth International Conference on Music Information Retrieval (ISMIR 2005)*, pages 34–41, 2005.

[17] Liping Liu and Thomas Dietterich. A conditional multinomial mixture model for superset label learning. In *Advances in Neural Information Processing Systems 25*, pages 557–565, 2012.

[18] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online learning for matrix factorization and sparse coding. *J. Machine Learn. Res.*, 11:19–60, 2010.

[19] L. Neal, F. Briggs, R. Raich, and X. Fern. Time-frequency segmentation of bird song in noisy acoustic environments. In *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing*, 2011.

[20] S. Pancoast and M. Akbacak. Bag-of-audio-words approach for multimedia event classification. In *INTERSPEECH*, 2012.

[21] Grigorios Tsoumakas, Eleftherios Spyromitros-Xioufis, Jozef Vilcek, and Ioannis Vlahavas. Mulan: A java library for multi-label learning. *Journal of Machine Learning Research (JMLR)*, 12:2411–2414, July 12 2011.

[22] Min-Ling Zhang. ML-RBF: RBF neural networks for multi-label learning. *Neural Processing Letters*, 29(2): 61–74, 2009.

[23] M.L. Zhang and Z.H. Zhou. Multi-instance clustering with applications to multi-instance prediction. *Applied Intelligence*, 31(1):47–68, 2009.