DCASE 2018 TASK 2: ITERATIVE TRAINING, LABEL SMOOTHING, AND BACKGROUND NOISE NORMALIZATION FOR AUDIO EVENT TAGGING

Thi Ngoc Tho Nguyen[1*], *Ngoc Khanh Nguyen*[2], *Douglas L. Jones*[3], *Woon Seng Gan*[1],

[1] Nanyang Technological University, Electrical and Electronic Engineering Dept., Singapore, {nguyenth003, ewsgan}@ntu.edu.sg
[2] SWAT, Singapore {kylenguyen}@swatmobile.io
[3] University of Illinois at Urbana-Champaign, Dept. of Electrical and Computer Engineering, Illinois, USA, {dl-jones}@illinois.edu

ABSTRACT

This paper describes an approach from our submissions for DCASE 2018 Task 2: general-purpose audio tagging of Freesound content with AudioSet labels. To tackle the problem of diverse recording environments, we propose to use background noise normalization. To tackle the problem of noisy labels, we propose to use pseudo-label for automatic label verification and label smoothing to reduce the over-fitting. We train several convolutional neural networks with data augmentation and different input sizes for the automatic label verification process. The label verification procedure is promising to improve the quality of datasets for audio classification. Our ensemble model ranked fifth on the private leaderboard of the competition with an mAP@3 score of 0.9496.

Index Terms— Audio event tagging, Background noise normalization, Convolutional neural networks, DCASE 2018, Label smoothing, Pseudo-label

1. INTRODUCTION

The FreesoundDataset (FSD) [1] is an open general-purpose and large-scale audio dataset with the aim to promote the advancement in audio research. The data are crowd-sourced and dynamically added into the dataset via the Freesound platform, where everyone can contribute his or her own records. Such method to collect data has several advantages such that the dataset can be developed into a large dataset with a great variety of audio contents, and researchers have full access to the raw audio wave files. However, the crowd-sourcing mechanism also introduces several challenges such that unverified labels, and a wide variability in recording devices, recording environments, and audio quality.

Task 2 (audio tagging) of Dcase 2018 challenge [2] explores some of the aforementioned challenges of the FSD. In Task 2, participants are asked to classify audio clips extracted from the FSD using a subset of labels from the AudioSet Ontology [3]. There are 41 labels that cover a wide range of sound activities such as musical instruments, human sounds, domestic sounds, and animals. The training set consists of 9473 audio clips with different lengths, out of which 3710 samples have manually-verified labels and 5763 samples have non-verified labels. This is an imbalanced dataset. The number of samples for each class varies from 94 to 300 samples per class. The test set consists of 9400 audio clips, out of which around 1600 samples are used to evaluate the system performance.

Two of the main challenges of Task 2 are the label noise and the diverse nature of crowd-sourced data. A popular approach for supervised learning using cross-entropy error with noisy labels is label smoothing [4, 5]. A network is over-confident when it places all probability on a single class in the training set [5]. Label smoothing instead assigns a value less than 1 to the target class and some value to other classes in the one-hot encoded label. This technique is equivalent to regularizing the network by penalizing lower entropy output distribution [5]. The audio tagging dataset has more than 60% non-verified labels that are automatically annotated, thus it is expected that there are many samples with incorrect labels. Because the number of non-verified samples is large compared to the size of the dataset, besides label-smoothing technique, we employ pseudo-label [6, 7], which is a widely used technique in semi-supervised learning. Pseudo-labeling technique iteratively assigns pseudo-labels to some unlabeled data and use those data together with labeled data in the next training iteration. To reduce the effect of the noisy label problem in Task 2 challenge, we propose to use the pseudo-labeling technique to iteratively and automatically verify the non-verified portion of the dataset. For those samples that the classified labels from the pseudo-labelling process are different from the non-verified labels, or the classification probabilities are below a certain threshold, we employ label smoothing because we are unsure if these samples are labeled incorrectly or they are from a different varieties of the target class.

Crowd-sourced data comes from different recording environments. For example, a telephone sound can be recorded in a quiet home or on a noisy street. These background noises introduce more variability to the signal and potentially reduce the performance of the learning algorithm as the model overfits to the background noise. A common approach to mitigate the problem of background noise is multi-condition training [8, 9] where different background noises are artificially added into the signals to simulate different environments. Multi-condition training can be interpreted as a data augmentation technique, thus the performance of the model depends on how many conditions that it is trained on. To reduce the effect of background noise in crowd-sourced data, we propose to normalize the audio signal by the background noise. The background noise normalization is inspired by the psychoacoustic and physiological observations that humans and other mammals dynamically adapt to the time-varying background noise level and selectively pay attention to sound signals that are above the noise level.

*This material is based on research work supported by the IAF-ICP: Singtel Cognitive and Artificial Intelligence Lab for Enterprises@NTU under the Research Theme on Edge Intelligence.

A common pipeline for audio event classification for small audio datasets such as ESC50 [10], Urbansoudn8K [11], and TUT Acoustic Scenes [12] includes feature extraction using log-mel energy, data augmentation using pitch-shift, time-stretch, block mixing [13], random erasing/cutout [14, 15], and classification using deep neural networks [16, 17]. The data set for Task 2 is slightly larger than the Urbansound8K and has more classes. In this paper, we will follow this general pipeline, and add in the background noise normalization to build a reliable ensemble to automatically verify the training data via pseudo-labeling. Task 2 is hosted on Kaggle website. The mAP@3 scores of our best ensemble on the public leaderboard (PLB)and the final private leaderboard (PrLB) are 0.9635 and 0.9496 respectively. Our team name is "Emilia_NTU", and we ranked fifth in the competition.

We organize the paper as follows. Section II shows the building blocks of a sound event tagging model. Section III presents the automatic label verification. We report the experimental results and our DCASE 2018 submissions in Section IV. Finally, we conclude the paper in Section V.

2. BUILDING A SOUND EVENT TAGGING MODEL

We use the following parameters to convert the provided mono audio files from the time domain to the frequency domain via short time Fourier transform (STFT): sampling rate of 44.1 kHz, length-1024 FFTs, Hanning window with 50% overlap.

2.1. Preprocessing

Our preprocessing step consists of silent removal and repeating short audio clip. For each audio sample, we use librosa [18] to trim the silent part of the audio. After that, we remove frames of which the energies fall below the 10^{th} percentile of all the frame energies of that sample. Experimental results show that silent removal improves accuracy of the model. One reason might be there are many segments within an audio clips that do not contain the sound of interest. For audio samples that are shorter than the required length for deep neural networks, we repeat the signal instead of zero padding. Experimental results show that signal repeating slightly improves the model performance.

2.2. Background noise normalization

Let $\mathbf{X}$ be the audio signal in STFT domain. $\mathbf{X}$ is a matrix of size $n\_ffts/2$ x n_frames, where n_ffts is the number of FFT points and n_frames is the number of time frame. For each audio clip, the local background noise $\mathbf{x_{noise}}$ of size $n\_ffts/2$ x 1 is estimated such that $\mathbf{x_{noise}(i)}$ is 10^{th} percentile of signal magnitude in frequency band i. For application with streaming or continuous input, background noise can be adaptively estimated and update for different segments of audio files [19]. The signal can be normalized by the background noise as

$$\mathbf{X_{bgnorm}} = \mathbf{X}/\mathbf{x_{noise}} \tag{1}$$

Fig. 1 show an example of background noise normalization for an audio clip labeled as *Chime*. After normalization, the background noise becomes Gaussian white noise, and the signals above the background noise are highlighted. Thus the proposed normalization is promising to reduce the effect of different background noises to the classification.

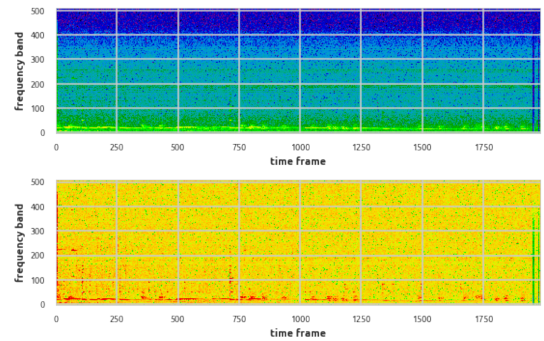

Figure 1: Example of background noise normalization for an audio clip with label *Chime*: Top is the unmodified STFT; Bottom is the normalized STFT

2.3. Feature extraction

We use librosa [18] to extract log-scale mel-spectrogram energy with the following parameters: maximum frequency of 18000 kHz and mel frequency filter bank of size 96.

2.4. Data augmentation

We use librosa [18] to generate the pitch-shift and time-stretch signal before training as the required processing time is long. The chosen pitch shift values in semitones are $[-2, -1, 1, 2]$. The chosen time-stretch ratios are [0.9, 0.95,1.05, 1.1]. We also augment data on-the-fly during training using mix-up [13], random erasing and cut-out [14, 15]. The data augmentation improves the performance of the algorithm, which is consistent with other researches [17].

2.5. Training model

We divide the provided training data into train (80%) and development set (20%). We use the competition evaluation metrics, mean average precision @ 3 (mAP@3) as the criteria to select network parameters. A convolutional neural network (CNN) that is a variant of VGG architecture [20] is used in our experiment. The CNN takes inputs as patches of log-mel spectrogram. The network architecture for input of size 128 frames and 96 mel bands is shown in Table 1. There is a total of 11 layers, which is similar to Vggish [21]. For training, we randomly extract patches from the log-mel spectrogram. The patches are normalized with the mean and standard deviation of all frames of all audio samples in the training set. For testing, we extract patches using a sliding window with hop size of 8 frames. The final prediction probability of an audio clip is the average prediction probabilities of all the extracted patches. To further improve the classification results, we extend the CNNs to take a second set of inputs which are the local mean and local standard deviation across all mel-frequency bands of each input patch. The mean and standard deviation is fed into two fully connected layers with 64 and 10 hidden neurons before being concatenated to the first fully connected layer of the CNN in Table 1.

3. AUTOMATIC LABEL VERIFICATION

The dataset contains diverse audio events with different lengths, such as short-duration gun shot and long-duration chime. To im-

Table 1: A model architecture for input of shape $(128, 96)$.

Stage	Output	Layers
Conv1	64x48	3x3, 24, BN, ReLU
		3x3, 24, BN, ReLU, maxpool, stride 2
Conv2	32x24	3x3, 48, BN, ReLU
		3x3, 48, BN, ReLU, maxpool, stride 2
Conv3	16x12	3x3, 48, BN, ReLU
		3x3, 48, BN, ReLU, maxpool, stride 2
Conv4	8x6	3x3, 64, BN, ReLU
		3x3, 64, BN, ReLU, maxpool, stride 2
Conv5	8x6	1x1, 64, BN, ReLU
fully connected	1x1	drop out, 128 fc, ReLU
		drop out, 41 fc, softmax
Number of parameters		5.48 x 10^5

prove the reliability of the label verification process, we build an ensemble of 3 CNN models. The three CNN models take patches of size $(128, 96)$, $(171, 96)$, and $(257, 96)$, which corresponds to audio segments of length 1.5 s, 2 s, and 3 s respectively. The second model use background noise normalized STFT to extract log-mel features. We combine the three models using their geometric mean which scores higher on the PLB compared to their arithmetic mean. We start the iterative training process with the verified labels. We only include the samples, of which the pseudo-labels are similar to the non-verified labels with classification probabilities greater than 0.5, to the training dataset for the next iteration. Our assumptions is that there are more samples with correct non-verified labels than those with incorrect labels. This assumption is justified since a model train on all the verified and non-verified samples produce a reasonable mAP@3 value on the PLB.

Table 2 shows the progress of the automatic label verification process using pseudo-label. The non-verified labels are considered as ground truth. We used mAP@3 as the evaluation metrics to compare the classified labels with the provided non-verified labels. The algorithm returns the best 3 classifications for each audio sample. mAP@3 returns a score of 1, $1/2$, and $1/3$, if the ground truth is matched with the best, the second best, and the third best classification respectively and returns a score of 0 otherwise. After the first iteration, our ensemble returns 3910 first best classifications, 537 second best classifications, 256 third best classifications, and 1060 incorrect classifications. Out of 3910 first best classification, 2680 samples have the classification probabilities greater than 0.5, thus they are added into the training set for the next iteration. We stop the process at iteration 4 when the number of incorrect classifications reaches a plateau. At iteration 4, we add 569 out of 576 first best classifications with the probabilities above 0.1. We relabel 152 samples out of 827 incorrect classifications, which have the classification probabilities greater than 0.5. The final training sets consists of 8039 verified samples and 1424 non-verified samples. Fig. 2 shows the histogram of the classification probabilities of the best classifications for iteration 1 and 4. At iteration 1, there are more "correct" non-verified samples, the ensemble returns high degree of confidence for many non-verified samples. At iteration 4, the distribution of the classification probabilities of the best guesses shifts toward 0. The left-over non-verified samples are either incorrectly annotated or contain different varieties of the audio class that the ensemble has not seen before. After the automatic verification process, for those 1424 left-over non-verified data, we use

Table 2: Iterative training for label verification.

# of iteration	Iter 0	Iter 1	Iter 2	Iter 3	Iter 4
# of verified samples	3710	3710	6390	7077	7470
# of new added samples	-	2680	687	393	569
# of correct pred at 1^{st} position	-	3910	1420	900	576
# of correct pred at 2^{nd} position	-	537	473	456	379
# of correct pred at 3^{rd} position	-	256	276	195	221
# of incorrect labels	-	1060	914	845	827
total # of non-verified labels	5763	5763	3083	2396	2003
mAP@3	-	0.740	0.567	0.498	0.419

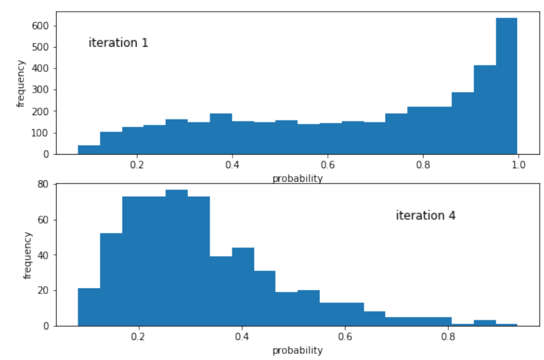

Figure 2: Histogram of classification probabilities of samples that the outputs of the pseudo-labelling ensemble are the same as the provided non-verified labels: Top is histogram for iteration 1; Bottom is histogram for iteration 4

label smoothing [22], which is defined as

$$y(i) = \begin{cases} \epsilon/k & \text{if i is none target} \\ 1 - (k-1)/k * \epsilon & \text{if i is target,} \end{cases} \quad (2)$$

in the subsequent training, where $y(i)$ is the one-hot encoded ground truth, k is the number of classes and ϵ is some small value. We sample ϵ from a uniform distribution between 0.1 and 0.3.

4. SUBMISSIONS FOR DCASE 2018 TASK 2

In this section, we discuss the results of our submissions to the Kaggle platform.

Model 1 was trained on all of the verified and non-verified data, with a segment length of 1.5 seconds, without silent removal, background normalization, data augmentation, label smoothing, and pseudo-label. Its mAP@3 values are 0.9125 and 0.8962 on the PLB and the PrLB respectively. This result shows that the non-verified data have many correct labels.

Model 2 was trained on all verified and non-verified data, with a segment length of 2 seconds, and background noise normalization. Ensemble of model 1 and 2 produced the mAP@3 values of 0.9286 and 0.9199 on the PLB and the PrLB respectively. This shows that background normalization and combining different segments lengths are helpful. In addition, the ensemble also generalizes better on the PrLB with less than 0.01 drop in the mAP@3 value. It is interesting that the mAP@3 values of model 2 on the PLB and the PrLB (0.8721 and 0.8559) are lower than those of model 1, however combining two models pushes the score significantly. From the observation that ensembles of diverse models normally perform

Table 3: Model properties.

Model ID	M1	M2	M3	M4	M5	M6	M7	M8
Input length (s)	1.5	2	1.5	1.5	2	3	1.5	2
Backgroun normalization	No	Yes	No	No	Yes	No	No	No
Silent removal	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Data augmentation	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Partially verified labels	No	No	No	Yes	Yes	No	No	No
Verified labels	No	No	No	No	No	Yes	Yes	Yes
Label smoothing	No	No	No	No	No	No	Yes	Yes
Additional input	No	No	No	No	No	No	No	Yes
Model mAP@3 (PLB)	0.913	0.872	0.936	0.924	0.894	0.949	0.932	0.930
Model mAP@3 (PrLB)	0.896	0.856	0.921	0.909	0.882	0.932	0.921	0.921
Ensemble mAP@3 (PLB)	0.913	0.929	0.951	-	-	0.964	0.962	0.963
Ensemble mAP@3 (PrLB)	0.896	0.920	0.936	-	-	0.947	0.948	0.950

better, we hypothesize that the background noise normalization provides some important emphasises of the signals that are not obvious in the non-normalized version.

Model 3 was trained on all verified and non-verified data, segment length of 1.5 seconds, with silent removal, data augmentation. The mAP@3 values of ensemble of model 1, 2 and 3 are 0.9507 and 0.9362 on the PLB and the PrLB respectively. This big jump on the PLB shows the importance of data augmentation and silent removal, which is consistent with the results from other researches that use small-medium size audio datasets [17, 23].

On the PrLB, our 3-model ensembles that are used for the automatically label verification process have the mAP@3 values of 0.8801, 0.9284, and 0.9222, and 0.9311 respectively at iteration 1, 2, 3 and 4. This shows that the more data we use for training, the better the validation score is. The mAP@3 value of the label-verification ensemble at iteration 3 with 7077 samples in the training set is lower than the mAP@3 value of the ensemble of model 1, 2, and 3. This implies that the learning algorithm can tolerate noisy labels and do better with more data.

The ensemble used for the label verification process consists of 3 models as mentioned in Section 3. We call these three models at iteration 4 as model 4, 5 and 6 respectively. The ensemble of model 1, 2, 3, 4, 5 and 6 has the mAP@3 values of 0.9640 and 0.9469 on the PLB. This shows that label verification using pseudo-labeling improves the mAP@3 score.

Model 7 was trained on re-labeled data after iteration 4 with a segment length of 1.5 seconds and label smoothing. Ensemble of model 1, 2, 3, 4, 5, 6, and 7 results in the mAP@3 values of 0.9623 and 0.9480 on the PLB and the PrLB respectively.

Model 8 was trained on re-labeled data after iteration 4 with a segment length of 2 seconds, label smoothing, and multiple inputs. The ensemble of model 1, 2, 3, 4, 5, 6, 7, and 8 returns the mAP@3 values of 0.9634 and 0.9496 on the PLB and the PrLB respectively which are the highest validation scores we obtained on the PLB and the PrLB. The last two ensembles do not increase the scores on the PLB but increases the score in the PrLB, which have more test data. This shows that label smoothing and multiple inputs are helpful in improving the generalization of the models.

Table 3 summaries the properties of 8 models that we used. The model mAP@3 rows report the mAP@3 value of each single model, while the ensemble mAP@3 rows report the mAP@3 value of the ensemble that includes all of the previous models. Fig. 3 shows the classification results of our 8 model ensemble. Overall, the ensemble performed reasonably well with a majority of classes have F1 scores above 0.85. There are several sound classes that achieve F1 score of 1The classes that have lowest F1 score are *Scissors*, *Chime*, *Squeak*, *Glockenspiel*, and *Firework*.

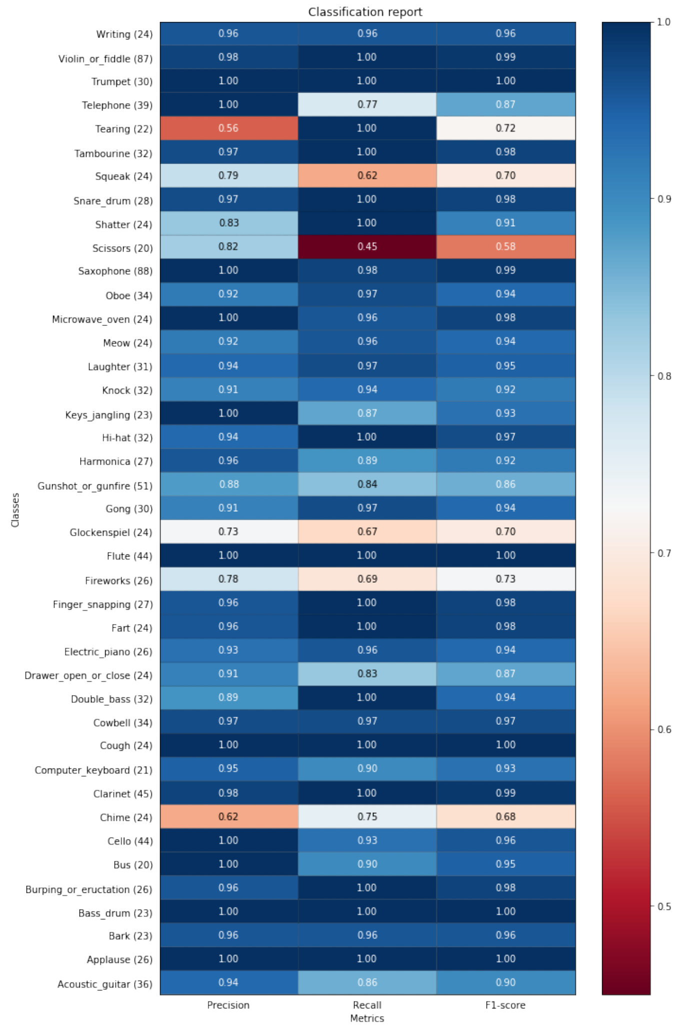

Figure 3: Classification results of an 8-model ensemble that ranked fifth in the competition

5. CONCLUSION

Datasets with high quality labels are crucial to supervised learning. However, manual annotation is expensive and time-consuming. The experimental results presented in this paper show that for small audio datasets, it is possible to increase the dataset size by training different models to automatically label new data. Manually annotation will be helpful for those ”difficult” samples that the models could not resolve. Thanks to pseudo-labelling, the number of samples that need manually annotation can be reduced significantly. The number of training samples that are incorrectly annotated was unknown at the time of the competition ended, however, we can observe that the learning models are quite robust to some degree of incorrect annotation. The proposed background noise normalization introduces a useful focus of the signal to the CNNs. In addition, it is beneficial to use different input lengths for different models to improve the ensemble accuracy on datasets with diverse audio events. In conclusion, the proposed approach shows meaningful improvement compared to the baseline system.

6. ACKNOWLEDGMENT

We would like to thank Daisuke Niizumi (@daisukelab) for valuable insights on topics of audio preprocessing, data augmentation, and model training on the competition's discussion forum.

7. REFERENCES

[1] E. Fonseca, J. Pons, X. Favory, F. Font, D. Bogdanov, A. Ferraro, S. Oramas, A. Porter, and X. Serra, "Freesound datasets: a platform for the creation of open audio datasets," in *Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR 2017)*, Suzhou, China, 2017, pp. 486–493.

[2] E. Fonseca, M. Plakal, F. Font, D. P. W. Ellis, X. Favory, J. Pons, and X. Serra, "General-purpose tagging of freesound audio with audioset labels: Task description, dataset, and baseline," 2018, submitted to DCASE2018 Workshop. [Online]. Available: https://arxiv.org/abs/1807.09902

[3] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 776–780.

[4] G. Pereyra, G. Tucker, J. Chorowski, Ł. Kaiser, and G. Hinton, "Regularizing neural networks by penalizing confident output distributions," *arXiv preprint arXiv:1701.06548*, 2017.

[5] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[6] D.-H. Lee, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Workshop on Challenges in Representation Learning, ICML*, vol. 3, 2013, p. 2.

[7] H. Wu and S. Prasad, "Semi-supervised deep learning using pseudo labels for hyperspectral image classification," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1259–1270, 2018.

[8] D. Garcia-Romero, X. Zhou, and C. Y. Espy-Wilson, "Multi-condition training of gaussian plda models in i-vector space for noise and reverberation robust speaker recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4257–4260.

[9] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 7398–7402.

[10] K. J. Piczak, "Esc: Dataset for environmental sound classification," in *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 1015–1018.

[11] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proceedings of the 22Nd ACM International Conference on Multimedia*, ser. MM '14. New York, NY, USA: ACM, 2014, pp. 1041–1044. [Online]. Available: http://doi.acm.org/10.1145/2647868.2655045

[12] A. Mesaros, T. Heittola, and T. Virtanen, "Tut database for acoustic scene classification and sound event detection," in *Signal Processing Conference (EUSIPCO), 2016 24th European*. IEEE, 2016, pp. 1128–1132.

[13] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[14] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.

[15] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," *arXiv preprint arXiv:1708.04896*, 2017.

[16] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *Machine Learning for Signal Processing (MLSP), 2015 IEEE 25th International Workshop on*. IEEE, 2015, pp. 1–6.

[17] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 279–283, March 2017.

[18] B. M. et. al., "librosa/librosa: 0.6.1," May 2018. [Online]. Available: https://doi.org/10.5281/zenodo.1252297

[19] R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statistics," *IEEE Transactions on speech and audio processing*, vol. 9, no. 5, pp. 504–512, 2001.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[21] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. Weiss, and K. Wilson, "Cnn architectures for large-scale audio classification," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017. [Online]. Available: https://arxiv.org/abs/1609.09430

[22] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1.

[23] S. Mun, S. Park, D. K. Han, and H. Ko, "Generative adversarial network based acoustic scene training set augmentation and selection using SVM hyper-plane," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017)*, November 2017, pp. 93–102.