On Open-Set Classification with L3-Net Embeddings for Machine Listening Applications

Kevin Wilkinghoff
Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE
Fraunhoferstraße 20, 53343 Wachtberg, Germany
kevin.wilkinghoff@fkie.fraunhofer.de

***Abstract*—Obtaining labeled data for machine listening applications is expensive because labeling audio data requires humans listening to recordings. However, state-of-the-art deep learning based systems usually require large amounts of labeled data to be trained with. A solution for this problem is to train a neural network with a large collection of unlabeled data to extract embeddings and then use these embeddings to train a shallow classifier on a small but labeled dataset suitable for the application. One example are Look, Listen, and Learn (L3-Net) embeddings, which are trained self-supervised to capture audio-visual correspondence in videos. Since shallow classifiers are trained discriminatively and thus tacitly assume a closed-set classification task, they do not perform well in open-set classification tasks. In this paper, a neural network that combines all L3-Net embeddings belonging to one recording into a single vector by using an x-vector mechanism as well as an open-set classification system based on that are presented. In experiments conducted on the open-set acoustic scene classification task belonging to the DCASE challenge 2019, the proposed system significantly outperforms a shallow discriminative classifier and all other previously published systems, while at the same time performing equally well as a shallow classifier on multiple closed-set machine listening datasets.**

***Index Terms*—open-set classification, deep audio embeddings, machine listening, acoustic scene classification, acoustic event classification**

I. Introduction

Deep learning based models are the state-of-the-art in machine listening, whose overall goal in an abstract sense is to somehow understand audio using machines. Labeling audio files and using convolutional neural networks (CNNs) as classifiers on spectral data is by far the most popular approach [1]–[5]. However, to train these models a large collection of labeled data is needed. But labeling data is expensive because it requires listening to recordings and thus needs experts and takes time. To solve this problem, a large dataset with unlabeled data can be used to train a neural network whose task it is to extract embeddings from audio data. These small dimensional embeddings should contain all information of the audio file needed for classifying it. Then, a shallow classifier can be trained on embeddings extracted from a small dataset with labeled data that is specifically chosen for a single machine listening application. A popular example are Look, Listen, and Learn (L^3-Net) embeddings [6]–[8] that are trained to capture audio-visual correspondace in between video frames and audio clips.

In open-set classification tasks [9], there are not only known classes to be recognized correctly (closed-set classification) but also unknown classes, which need to be marked as "unknown" by the system (outlier detection [10]). The major difficulty is that not all possible unknown classes are known a priori. Thus, for such classes no samples can be provided when training, but instances of that class still need to be rejected when testing. In conclusion, open-set classification problems are much more difficult to solve than closed-set classification problems when the same number of known classes is encountered. However, for machine listening applications in a realistic environment open-set problems are much more common since it can only very rarely be ensured that all possible sounds that may occur when running the system are previously known. Since shallow classifiers are trained to discriminate among the known classes and thus assume that data belongs to one of these known classes, they perform poorly when trying to detect outliers. Hence, they are not a suitable backend for L^3-net embeddings in open-set classification settings.

The contributions of this paper are the following: First, a network for L^3-net embeddings that combines all embeddings belonging to one audio file into a single vector is presented. On multiple datasets, it is shown that this network provides a similar performance than the standard shallow MLP classifier directly trained on L^3-net embeddings. Second, based on this network an open-set classification system for machine listening applications is presented. In additional experiments conducted on the open-set acoustic scene classification dataset of the Detection and Classification of Acoustic Scenes and Events (DCASE) challenge 2019 [11], it is shown that the proposed system significantly outperforms a shallow classifier and all other published systems for open-set classification.

II. L^3-Net Embeddings Backends

A. Look, Listen, and Learn (L^3-Net) embeddings

The goal of Look, Listen and Learn (L^3-Net) [6] is to detect audio-visual correspondence between a single video frame and an audio clip with a duration of 1s. Its overall structure is depicted in Fig. 1. The L^3-net consists of a video subnetwork and an audio subnetwork, which both consist of four convolutional blocks with pooling operations and extract an embedding from a video frame and audio clip, respectively. To check the correspondence between both embeddings, a fusion network, which concatenates both embeddings and uses

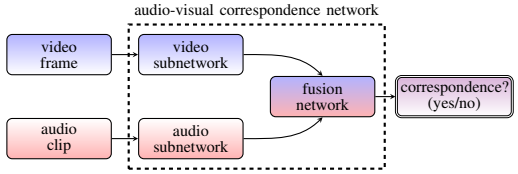

Fig. 1. Structure of the L^3-net for checking audio-visual correspondence.

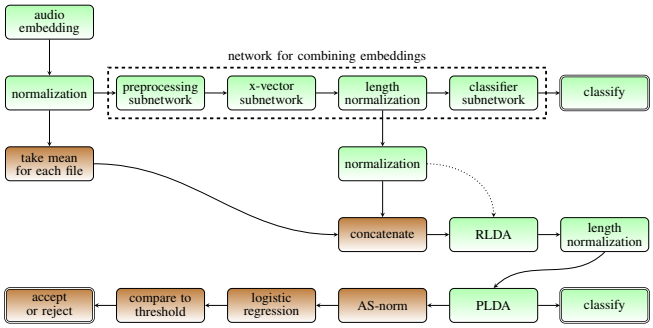

Fig. 2. Structure of the proposed L^3-net embedding backend. The blocks colored in green are needed for closed-set classification. Brown blocks only need to be considered for open-set classification.

two fully connected layers as well as a softmax layer for binary classification, is used.

Training the L^3-net can be done without annotated data and only requires a (large) dataset of videos. Positive examples consist of video frames and the corresponding audio clips from the same video. Negative examples can be provided by video frames and audio clips from different videos. After training, embeddings can be extracted using only the audio subnetwork.

In this paper, a pretrained model (openL3) from [8] is used to extract L^3-net embeddings. The model is pretrained on the music subset of AudioSet [12] and extracts 512 dimensional embeddings from overlapping windows with 1s length and a 0.1s hop size of Mel-spectrograms with 256 Mel bins. For additional details of openL3, the reader is referred to [8]. Throughout the paper, all embeddings are normalized by subtracting the mean and dividing by the standard deviation of all embeddings belonging to the task-specific training set.

B. Standard closed-set classification backend

A simple model for classifying audio data using L^3-net embeddings is a shallow Multilayer perceptron (MLP) as presented in [8]. It is trained discriminatively for classifying single embeddings. The MLP consists of two fully connected layers of size 512 and 128 with Rectified Linear Units (ReLU) as propagation functions and an output layer with a softmax nonlinearity for classification, whose size corresponds to the total number of classes. To classify an entire audio file, the mean per class of all resulting scores of the embeddings belonging to that file is taken. Using this mean, a class can be predicted for each file via maximum likelihood.

C. X-vector based closed-set classification backend

Open-set classification problems can be decomposed into two subproblems [13]: 1) outlier detection and 2) closed-set classification. Thus, it is vital that the closed-set performance of an open-set classification backend for L^3-net embeddings is not worse than the performance obtained with a shallow MLP. In this subsection, the backend for closed-set classification, which will be extended to an open-set classification system, will be presented. Both are depicted in Fig. 2.

TABLE I
ARCHITECTURE OF THE NETWORK FOR COMBINING EMBEDDINGS.

Subnetwork	Layer	Output Shape
Preprocessing	Input	(T, 512)
	Batch normalization	(T, 512)
	Gaussian noise (standard deviation: 0.1)	(T, 512)
X-vector	1D Convolution (kernel size=3, Leaky ReLU: 0.1)	(T, 256)
	Batch normalization	(T, 256)
	1D Convolution (kernel size=3, Leaky ReLU: 0.1)	(T, 256)
	Batch normalization	(T, 256)
	1D Convolution (kernel size=5, Leaky ReLU: 0.1)	(T, 256)
	Batch normalization	(T, 256)
	1D Convolution (kernel size=1, Leaky ReLU: 0.1)	(T, 256)
	Batch normalization	(T, 256)
	1D Convolution (kernel size=1, Leaky ReLU: 0.1)	(T, 512)
	Batch normalization	(T, 512)
	Mean	512
	Standard deviation	512
	Concatenation	1024
	Dense (Linear)	256
	Length normalization	256
Classifier	Gaussian noise (standard deviation: 0.1)	256
	Leaky ReLU: 0.1	256
	Batch normalization	256
	Dropout (rate: 0.8)	256
	Dense (Leaky ReLU: 0.1)	256
	Batch normalization	256
	Dropout (rate: 0.5)	256
	Dense (Leaky ReLU: 0.1)	128
	Batch normalization	128
	Dense (Softmax)	#Classes

The underlying idea is to combine all embeddings belonging to a single audio file into a single embedding, which is well suited to discriminate among the classes that appear in a specific machine listening application. One way to do this is to use so-called x-vectors [14], which are the state-of-the-art in speaker recognition and are usually trained on Mel-frequency cepstral coefficients (MFCCs) [15]. The x-vector subnetwork of the proposed system consists of five convolutional layers in time and a statistics pooling layer that outputs the concatenation of mean and standard deviation as a another embedding called x-vector. To be clear, here the x-vector subnetwork gathers statistics of L^3-net embeddings instead of MFCCs. A classifier subnetwork is used to incorporate a discriminative behavior into the x-vectors tailored towards the classes of the dataset. Tab. I contains all details of the network's architecture.

Training the network is done by minimizing the categorical crossentropy with Adam [16] and a batch size of 32. To effectively increase the amount of training data and prevent the network from overfitting, the following data augmentation techniques are used while training. First of all, mixup [17], which is a linear interpolation between two randomly chosen training samples, with a mixing coefficient drawn from a uniform distribution are used. For acoustic event detection, SpecAugment [18] without time warping, i.e. only frequency and time masking, has been successfully applied to spectrograms [19]. This is the reason why all L^3-net embeddings are masked with 5 frequency masks of size 64 and 10 time masks both of size 64. Note that L^3-net embeddings do not (directly) contain frequency information and the term frequency mask is only kept for convenience. Additionally, random circular

shifts of up to 99% in time are applied. To speed up and stabilize the training process, batch normalization layers [20] are present throughout the network. For regularization, dropout [21], Gaussian noise layers with a standard deviation of 0.1 and L2-regularization with a weight of 0.00001 are used. Additionally, the gradient is weighted inversely proportional to the number of training samples per class to ensure that the network is not biased towards classes with more training samples. All neural networks are implemented using Keras [22] with Tensorflow [23].

After combining all L^3-net embeddings of all files into x-vectors, a standard classification chain for x-vectors can be applied. As a first step, the x-vectors are normalized by subtracting the mean and dividing by the standard deviation of all x-vectors belonging to the training set. Next, linear discriminant analysis (LDA) is applied. The goal of LDA is to reduce the x-vectors' dimensionality by projecting them onto a subspace suitable for discriminating among the classes. Using LDA, the maximum dimension of the subspace is the number of classes minus 1. Regularized linear discriminant analysis (RLDA), as described in [24], adds a small number to the diagonal of the between-class and and within-class covariance matrices, thus turning them into full rank. This allows an arbitrary dimension to be chosen for the subspace. For all experiments, the LDA dimension has been optimized to reduce the error rate. After that, all LDA projected x-vectors are length-normalized again. As a last step, a two-covariance probabilistic linear discriminant analysis (PLDA) model [25], [26] as implemented in [27] is trained. Let y denote a class model, which is a vector with the same dimension as an x-vector, and ϕ denote an x-vector belonging to that class. Then, the two-covariance PLDA model is described by the following equations:

$$\begin{aligned} y &\sim \mathcal{N}(y|\mu, B^{-1}) \\ \phi|y &\sim \mathcal{N}(\phi|\mu, W^{-1}) \end{aligned} \tag{1}$$

where μ is the class mean, B^{-1} the inter-class covariance matrix and W^{-1} the intra-class covariance matrix. Using this model, a log-likelihood ratio can be computed that compares the likelihoods of two x-vectors belonging to the same class or belonging to different classes. Using mean x-vectors for each class derived from the training data, one can obtain likelihoods for each class and decide to which of these classes an x-vector belongs to by using maximum likelihood.

D. Open-set classification backend

The overall goal of this paper is to present an open-set classification backend for L^3-net embeddings. The PLDA model returns log-likelihood ratios between the null hypothesis that two x-vectors belong to the same class versus the alternative hypothesis that both belong to different classes. Hence, one can easily use a fixed threshold for the resulting scores and output the class "unknown" whenever all scores are below that chosen threshold. The same procedure can be used for the softmax output of the MLP and the x-vector network. However, as the softmax function models a posterior probability and tacitly assumes that each test file belongs to one of the classes, this is not expected to result in a good outlier detection performance.

Another way to combine the L^3-net embeddings is to take the mean embedding of each file. The resulting mean embeddings can then be handled the same way as x-vectors. The major difference to the x-vectors is that the resulting mean is not trained discriminatively and not adapted to the small dataset of the application. Therefore, the mean embeddings are not expected to perform as well as the x-vectors for closed-set classification. But since discriminative behavior among the known classes is not useful when detecting outliers and mean embeddings have an entirely different view on the data than x-vectors, combining them seems to be beneficial. This is achieved by simply concatenating both, x-vectors and mean embeddings, before applying RLDA and PLDA.

To detect outliers, adaptive symmetric normalization (AS-norm) [28], [29] is applied to the scores resulting from the PLDA model because this improves the performance in open-set settings [30], [31]. AS-Norm makes use of a set of files, called cohort, to normalize the scores and is defined as

$$\begin{aligned} s(e,t)_{\text{as-norm}} := \frac{1}{2}\Bigg(&\frac{s(e,t) - \mu(s(e, \mathcal{C}_{\text{top}(e,n_1)}))}{\sigma(s(e, \mathcal{C}_{\text{top}(e,n_1)}))} \\ &+ \frac{s(e,t) - \mu(s(t, \mathcal{C}_{\text{top}(t,n_2)}))}{\sigma(s(t, \mathcal{C}_{\text{top}(t,n_2)}))} \Bigg) \end{aligned} \tag{2}$$

where $s(e,t)$ denotes the score between mean x-vector e and test x-vector t, and μ and σ denote mean and standard deviation of the scores, respectively. Furthermore, $\mathcal{C}_{\text{top}(e,n_1)}$ denotes the $n_1 \in \mathbb{N}$ samples $\{c_k \in \mathcal{C} : k = 1, ..., n_1\}$ of the cohort $\mathcal{C}$ with highest scores $s(e, c_k)$ ($\mathcal{C}_{\text{top}(t,n_2)}$ is defined analogously). In the presented system, the cohort consists of all training files belonging to known classes and $n_1 = 900$ and $n_2 = 6500$ have been used as cohort sizes. These values have been determined by maximizing the performance on the validation set. Finally, the scores are calibrated with logistic regression, as implemented in [32], and a threshold of 0.5 is used to mark files as "unknown" whenever a score is below that threshold.

III. Experimental results

A. Comparison with standard MLP backend

As a first experiment, the performance of the proposed backend at various output stages will be compared to the one obtained with the shallow MLP presented in [8]. This is done to ensure that the x-vector based backend does not perform worse than a shallow discriminative classifier in closed-set classification tasks. The following three datasets are being used for that purpose: UrbanSound8k [33], which contains urban sound events, ESC-50 [34], containing environmental sound events, and DCASE 2013 SCD [35], which contains acoustic scenes. More details about the datasets can be found in Tab. II. Note that the length of audio files in Urbansound8k varies. Although the x-vector network is capable of dealing with varying input sizes, short audio files have been repeated until a length of 4s is reached.

TABLE II
DATASETS BEING USED FOR COMPARING BACKENDS.

Dataset	#Files	File Length	#Classes	Evaluation
UrbanSound8k [33]	8732	≤ 4 seconds	10	10 cross-validation folds
ESC-50 [34]	2000	5 seconds	50	5 cross-validation folds
DCASE 2013 SCD [35]	200	30 seconds	10	train and test set
DCASE 2019 Task 1C [11]	17050	10 seconds	10+?	development and leaderboard set

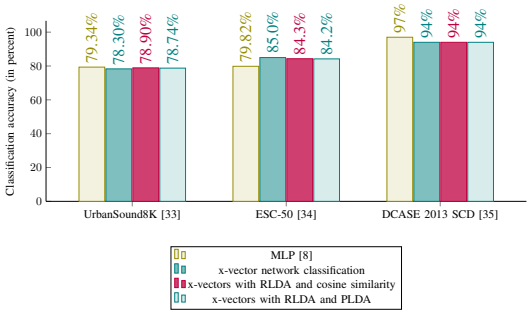

Fig. 3. Comparison of classification accuracies obtained with different backends. The results for the MLP are taken from [8].

The closed-set classification results can be found in Fig. 3. First of all, it can be seen that the performances of the presented network for extracting x-vectors itself are very close to the ones obtained with RLDA and PLDA applied to the x-vectors for all three datasets. When classifying with cosine similarity, the mean of all x-vectors belonging to a single class has been used as the reference vector. Since log-likelihood ratios perform better than the softmax output of discriminative networks, only PLDA will be considered in the open-set classification experiments.

Second and more importantly, the performance of the MLP is roughly the same as the one obtained with x-vectors. But there are huge differences among the three datasets. For Urbansound8k, the performance of all approaches is very close, but the MLP performs sightly better than the x-vector based systems. The same is true for DCASE 2013 SCD, which is a very small dataset and only consists of 100 training and 100 test files. On ESC-50, the performance of the x-vector based systems are significantly better than the shallow MLP. A probable reason is that ESC-50 contains much more classes (50) than both of the other datasets, which only contain 10 classes. In sum, an x-vector based system is a suitable closed-set classification backend for L^3-net embeddings.

B. Open-set acoustic scene classification

The open-set classification performance is evaluated with the dataset belonging to the open-set acoustic scene classification task of the DCASE challenge 2019 [11]. The dataset consists of a development set, which is separated into a training and validation split, a test set, whose labels have not been published, and two leaderboard sets, which are still available on Kaggle and thus are used for the experiments[1]. More details about the dataset can be found in Tab. II. The major difference to the closed-set classification datasets is that

[1]See https://www.kaggle.com/c/dcase2019-task1c-leaderboard/leaderboard

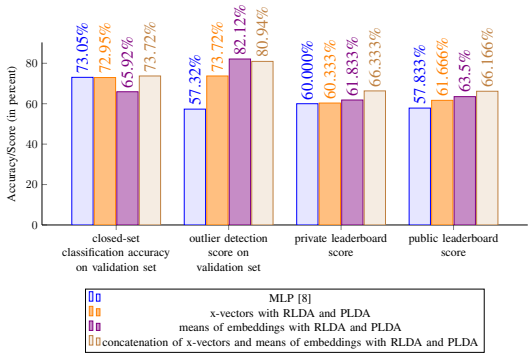

Fig. 4. Comparison of open-set classification scores obtained with different backends on the DCASE 2019 Task 1C dataset [11].

there are a 10 known classes and an unknown number of additional classes that the system has to mark as "unknown".

The results can be found in Fig. 4. First of all, the x-vector backend has a slightly higher closed-set classification accuracy than the simple MLP backend [8]. As emphasized before, the shallow MLP is not expected to perform well when detecting outliers. The experimental results validate this well known fact since the MLP has a much lower outlier detection score on the validation set than the x-vector backend.

When taking the mean of all L^3-net embeddings belonging to a single audio file, one can see that this yields a significantly worse closed-set classification accuracy but a significantly higher outlier detection score than the x-vectors. Since the x-vectors have been trained discriminatively and obtaining the mean embeddings requires no training at all, these results were to be expected. To have all information available in a single vector, the x-vectors and corresponding mean embeddings have been concatenated. As a result, these concatenations significantly outperform both individual components in terms of closed-set classification accuracy as well as outlier detection score. Moreover, both scores are higher than any other submitted system on the Kaggle leaderboards belonging to the task despite the fact that no ensembling techniques have been used for the presented system. Nevertheless, the results clearly show that L^3-net embeddings in combination with the presented backend are a very powerful tool for open-set machine listening applications.

IV. CONCLUSIONS AND FUTURE WORK

In this work, an open-set classification system for machine listening applications based on a neural network for combining L^3-net embeddings into x-vectors has been presented. On multiple datasets, this system performs equally well as a shallow MLP, directly trained to classify L^3-net embeddings, when used for closed-set classification. Moreover, the presented system significantly outperforms the MLP as well as all previously published results on the open-set acoustic scene classification task of the DCASE challenge 2019.

In the near future, additional experimental evaluations of the proposed system for open-set classification problems that are part of the DCASE challenge 2020 will be carried out to

complement the results presented in this paper. Furthermore, it is planned to experiment with ensembles of L3-net embeddings and more traditional CNNs, which are directly trained on the small dataset belonging to the task at hand. This should improve the closed-set classification performance significantly due to the totally different nature of both approaches. Additional investigations regarding the influence of the number of classes on the closed-set classification performance will also be undertaken, since the proposed backend performed significantly better then the shallow classifier on ESC50, which contains 50 instead of 10 classes.

REFERENCES

[1] Karol J. Piczak, "Environmental sound classification with convolutional neural networks," in *International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2015, pp. 145–150.

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

[3] Kevin Wilkinghoff, "General-purpose audio tagging by ensembling convolutional neural networks based on multiple features," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*. 2018, pp. 44–48, Tampere University of Technology.

[4] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yong Xu, Wenwu Wang, and Mark D. Plumbley, "Cross-task learning for audio tagging, sound event detection and spatial localization: DCASE 2019 baseline systems," *arXiv preprint arXiv:1904.03476*, 2019.

[5] M. N. Istiaq Ahsan, Csaba Kertész, Annamaria Mesaros, Toni Heittola, Andrew Knight, and Tuomas Virtanen, "Audio-based epileptic seizure detection," in *27th European Signal Processing Conference (EUSIPCO)*, 2019, pp. 1–5.

[6] Relja Arandjelovic and Andrew Zisserman, "Look, listen and learn," in *International Conference on Computer Vision (ICCV)*. IEEE, 2017, pp. 609–617.

[7] Relja Arandjelovic and Andrew Zisserman, "Objects that sound," in *European Conference on Computer Vision (ECCV)*, 2018, pp. 435–451.

[8] Jason Cramer, Ho-Hsiang Wu, Justin Salamon, and Juan Pablo Bello, "Look, listen, and learn more: Design choices for deep audio embeddings," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3852–3856.

[9] Walter J. Scheirer, Anderson Rocha, Archana Sapkota, and Terrance E. Boult, "Towards open set recognition," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, July 2013.

[10] Charu Aggarwal, *Outlier Analysis*, Springer, 2nd edition, 2017.

[11] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen, "Acoustic scene classification in DCASE 2019 challenge: Closed and open set classification and data mismatch setups," in *Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*. 2019, pp. 164–168, New York University.

[12] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[13] Kevin Wilkinghoff and Frank Kurth, "Open-set acoustic scene classification with deep convolutional autoencoders," in *Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*. 2019, pp. 258–262, New York University.

[14] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[15] Steven Davis and Paul Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[16] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations (ICLR)*, 2015.

[17] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz, "Mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations (ICLR)*, 2018.

[18] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2019.

[19] Wootaek Lim, "SpecAugment for sound event detection in domestic environments using ensemble of convolutional recurrent neural networks," in *Detection and Classification of Acoustic Scenes and Events Workshop (DCASE2019)*, 2019, pp. 129–133.

[20] Sergey Ioffe and Christian Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, 2015, vol. 37, pp. 448–456.

[21] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[22] François Chollet et al., "Keras," https://keras.io, 2015.

[23] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al., "Tensorflow: A system for large-scale machine learning," in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2016, pp. 265–283.

[24] Hossein Zeinali, Lukas Burget, and Jan Cernocky, "Convolutional neural networks and x-vector embedding for DCASE2018 acoustic scene classification challenge," in *Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*. 2018, pp. 202–206, Tampere University of Technology.

[25] Simon J.D. Prince and James H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *11th International Conference on Computer Vision. ICCV*. IEEE, 2007, pp. 1–8.

[26] Niko Brümmer and Edward De Villiers, "The speaker partitioning problem.," in *ODYSSEY - The Speaker and Language Recognition Workshop*, 2010, pp. 202–209.

[27] Aleksandr Sizov, Kong Aik Lee, and Tomi Kinnunen, "Unifying probabilistic linear discriminant analysis variants in biometric authentication," in *Proc. S+SSPR*. Springer, 2014, pp. 464–475, Software available at https://sites.google.com/site/fastplda/.

[28] Sandro Cumani, Pier Domenico Batzu, Daniele Colibro, Claudio Vair, Pietro Laface, and Vasileios Vasilakakis, "Comparison of speaker recognition approaches for real applications," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2011, pp. 2365–2368.

[29] Zahi N Karam, William M Campbell, and Najim Dehak, "Towards reduced false-alarms using cohorts," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2011, pp. 4512–4515.

[30] Roberto Font, "A denoising autoencoder for speaker recognition. results on the MCE 2018 challenge," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6016–6020.

[31] Elie Khoury, Khaled Lakhdhar, Andrew Vaughan, Ganesh Sivaraman, and Parav Nagarsheth, "Pindrop labs' submission to the first multi-target speaker detection and identification challenge," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2019, pp. 1502–1505.

[32] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[33] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *22nd ACM International Conference on Multimedia*, Orlando, FL, USA, 2014, pp. 1041–1044.

[34] Karol J. Piczak, "ESC: Dataset for environmental sound classification," in *23rd ACM international conference on Multimedia*. ACM, 2015, pp. 1015–1018.

[35] Dan Stowell, Dimitrios Giannoulis, Emmanouil Benetos, Mathieu Lagrange, and Mark D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Transactions on Multimedia*, vol. 17, no. 10, pp. 1733–1746, 2015.