2019 IEEE International Conference on Multimedia and Expo (ICME)

ACOUSTIC SCENE CLASSIFICATION WITH MISMATCHED RECORDING DEVICES USING MIXTURE OF EXPERTS LAYER

Truc Nguyen [*], *Franz Pernkopf* [†]

Graz University of Technology,
Signal Processing and Speech Communication Lab.,
Inffeldgasse 16c, A-8010 Graz, Austria/Europe,
{t.k.nguyen, pernkopf}@tugraz.at

ABSTRACT

Recently, a mismatch in acoustic conditions such as a temporal recording gap as well as different recording devices for the development and the evaluation data has been considered in Acoustic Scene Classification (ASC). This brings ASC closer to real world conditions. In this paper, we address ASC with mismatching recording devices. This has been introduced as task 1B of the DCASE 2018 challenge. We proposed a flexible and robust model that uses a mixture of experts (MoE) layer replacing the fully connected dense layer such that each expert can adapt to the specific domains of the data. Furthermore, we observe different Convolutional Neural Network (CNN) models as well as the number of the experts of the MoE dense layer using log-mel features. In addition, we perform mixup data augmentation to enhance the robustness of our models. In experiments, the classification performance is 66.1% using 15 experts in the MoE dense layer with approximately 2M parameters. This outperforms the best model of task 1B of the DCASE 2018 challenge by 2.5% (absolute). This model uses an ensemble selection of 12 individual models with ∼12M parameters.

Index Terms— Acoustic scene classification, convolutional neural network, mixture of experts layer, mixture of softmaxes.

1. INTRODUCTION

Acoustic scene classification (ASC) is a multi-class classification task classifying the recorded environment sounds as specific acoustic scenes that characterize either the location or situation such as park, metro station, tram, etc. It has been a task in the Detection and Classification of Acoustic Scenes and Events (DCASE) challenges providing the largest publicly available data sets for ASC.

Compared to ASC tasks of DCASE 2013 and DCASE 2016, the difficulty has been increased for DCASE 2018. Beside providing shorter segments of 10 s of audio data, there are mismatches between the development data set and the evaluation set. According to [1], there was a mismatch in acoustic conditions in the evaluation and the development set of the DCASE 2017 challenge i.e. data sets were recorded in similar locations with the same device but almost one year later. This temporal gap is the reason of a significant drop in performance of all systems. The DCASE 2018 challenge introduced task 1B with data sets recorded by four different devices in 6 different cities in Europe instead of only one city. This causes even more mismatch in the data. Especially, a part of the evaluation set is a compressed version of recorded audio data from device D that is not included in the development data set. This causes an extreme mismatch in the DCASE 2018 challenge data.

Recent ASC research mostly uses log-mel energies and mel-frequency cepstral coefficients (MFCC) as features. Beside that, harmonic-percussive source separation (HPSS) and I-vectors extracted from these mel-frequency scales have been effective features contributing to the success in the last DCASE challenges [2], [3], [4]. Some systems use the raw waveform [5] and conventional signal processing methods such as wavelet decomposition [6], [7] for feature extraction. For classification, deep learning (DL) has been the preferred solution. Beside well-known DL models used for image databases such the VGGNet [2], [3], [4], and Xception [8], popular models for acoustic data such as Recurrent Neural Networks (RNNs) or Long Short term Memmories (LSTMs) have been used [9], [10], [11]. Recently, attention mechanisms have been introduced [12], [13], [14] to supplement vanilla DL models. In addition, techniques of data augmentation such as Generative Adversarial Networks (GANs) [15] and mixup have been used [3]. Furthermore, ensemble methods helped the systems to the top performances in DCASE 2017 [2] and DCASE 2018 [3], [16].

Although a variety of ASC systems have been proposed, there is a limited number that focused on the analysis of the mismatching acoustic conditions. In this paper, we focus on the DCASE 2018 data of task 1B where the recording took place at several cities with different devices. We propose a

[*]Thanks to Vietnamese - Austrian Government Scholarship for funding.
[†]Thanks to Austrian Science Fund.

978-1-5386-9552-4/19/$31.00 ©2019 IEEE
DOI 10.1109/ICME.2019.00287

robust model that includes many experts modelling specific aspects in the feature spaces. This approach has been successful for the tasks of language modeling and machine translation where the feature spaces are much larger compared to that of ASC, [17], [18], [19]. We feed log-mel energies to various Convolutional Neural Network (CNN) structures where a mixture of experts layer is introduced as component of the CNN models. We replaced the fully connected dense layer by the mixture of experts layer. In addition, mixup data augmentation is applied to leverage the performance of our system.

The rest of paper is organized as follows. Related work for mixture of experts is introduced in Section 2. Section 3 presents the proposed ASC system, including the audio preprocessing, the mixture of experts layer, a variety of CNN structures and mixup data augmentation. In Section 4, we provide experiments and evaluate the performance of the proposed approach. Section 5 concludes the paper.

2. RELATED WORK FOR MIXTURE OF EXPERTS

The mixture of experts layer proposed in this work is inspired from the long-existing idea called Mixture of Experts (MoE) in 1991 [20]. Both the MoE and the MoE layer are composed of a set of modules referred to as expert networks which are suitable to model various regions of the input space. A gating network addresses the suitable expert for each input region. According to related work of MoE in [18], a MoE is one model that was introduced by different types of expert architectures such as support vector machines, Gaussian processes and deep networks, while a MoE layer is a part of a deep model that can be any specific layer in a network [18], [19]. These systems have been useful for language modeling and machine translation tasks.

There are different structures of an MoE layer related to the gating network and expert networks. For example, the sparsely-gated MoE layer of Shazeer et al. [18] embedded within a recurrent language model. They propose a noisy top-K gating instead of using a softmax gating by adding sparsity and noise before the softmax function. It leads to a computational benefit as well as it allows for training of a very large network architecture including up to thousand experts. In addition, Yang et al. [19] considered a MoE layer as a mixture of contexts (MoC). This is located before the output layer in the network model; beside that, they mainly proposed a mixture of softmax (MoS) layer replacing the softmax function in the output layer to break the softmax bottleneck where activations of experts and gating networks are softmax. Furthermore, Orhan [21] observed that the softmax bottleneck is a special case of a more general degeneracy problem that can happen even when a mapping of input spaces and output spaces of the MoE layer has the same dimensionality and even when the nonlinearity is not a softmax function. To prove this, they proposed a MoE layer that is similar to the MoS but where the softmax activation of the experts is replaced by ReLU activations and no Tanh activation for the linear combination of the inputs is performed.

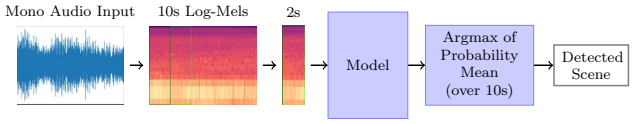

Fig. 1. Proposed System.

3. PROPOSED ARCHITECTURE

The proposed system is illustrated in Fig.1. The system consists of three stages. First, the mono audio signals are converted to time-frequency representation and then split into 2s segments. These features are fed to a CNN model with a dense MoE layer replacing the traditional fully connected dense layer. Finally, the probability outputs of the CNN are averaged over 5 continuous 2s segments and then the argmax operation is performed to obtain the final label predictions.

3.1. Audio pre-processing

This system is using the DCASE 2018 task 1 data set which is recorded with different recording devices at different cities. We keep the sampling rate at 44.1 kHz. (Since)the audio segments are only 10 s. We extract 128 bin mel energies of the provided mono audio such that obtain the spectral characteristics of the data. The window function of the short-time Fourier transform (STFT) is a Hann window and the window size is selected as 40ms with 20ms hop size.

We use only log-mel energies for to single input (SI) CNNs. They are more efficient than multiple inputs (MI) CNNs where a pair of the mel spectrogram and its nearest neighbor filtered version are used for task 1B of DCASE 2018 [16]. Furthermore, based on the results of the best systems in DCASE 2017 and 2018, and our experimental results, we can see that processing acoustic scenes in short segments is better than using the entire segments. We split audio segments into 2 s samples (128 bins x 100 frames per sample). All features are converted into logarithmic scale and normalized to zero mean and unit variance.

3.2. Mixture of Experts Layers

There are two types of the MoE layer introduced in [21], namely, the dense MoE layer and the convolutional MoE layer. They are mathematically defined in eq.1 and eq.2, i.e.

$$\mathbf{y} = \sum_{k=1}^{K} \underbrace{g(\mathbf{V}_k^T \cdot \mathbf{z} + \mathbf{b}_k)}_{G_k(\mathbf{z})} \cdot \underbrace{f(\mathbf{W}_k \cdot \mathbf{z} + \mathbf{c}_k)}_{E_k(\mathbf{z})}, \qquad (1)$$

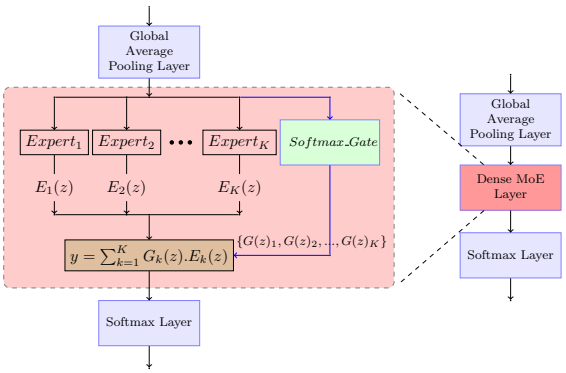

Fig. 2. A Mixture of Experts (MoE) layer embedded within a ASC model.

$$\mathbf{y} = \sum_{k=1}^{K} \underbrace{g(\mathbf{V}_k^T \cdot \mathbf{z} + \mathbf{b}_k)}_{G_k(\mathbf{z})} \cdot \underbrace{f(\mathbf{W}_k * \mathbf{z} + \mathbf{c}_k)}_{E_k(\mathbf{z})}, \qquad (2)$$

where K denotes the number of experts, $g(\cdot)$ is the softmax gating, $f(\cdot)$ and $f(*)$ are the experts using ReLU activation with a linear operation for dense MoE layers and a convolution operation for convolutional MoE layers, respectively. $\mathbf{V}_k$, $\mathbf{b}_k$ and $\mathbf{W}_k$, $\mathbf{c}_k$ denote the weights and the bias of the gating function and the expert k, respectively, while $\mathbf{z}$ is the input vector of the gating and expert function.

The mixture of experts layer is illustrated as a red block in the Figure 2. The dense and convolutional MoE layers can be used in the same way as the corresponding dense and convolutional layers[1].

In this work, we use the dense mixture of experts layer as a fully connected layer between the global average pooling and the output (softmax) layer. There are several experts in the dense MoE layer and each expert is considered as a component consultant corresponding to specific features extracted from previous layers of a CNN model. These experts enable to adapt to a diversity of extracted features from the different recording devices and as a result, it enhances the performance of the model compared to using only a fully connected dense layer. Figure 2 shows the structure of a MoE layer embedded in our ASC system.

In addition, we also tried to replace the classical convolutional layers by the convolutional MoE layers of one CNN model using two single CNN blocks as a feature extractor. We empirically observed that the convolutional MoE layers are not helpful for the system in terms of accuracy and number of parameters as well as computation time.

[1]The source code of the MoE layers are provided by Emin Orhan in https://github.com/eminorhan/mixture-of-experts.

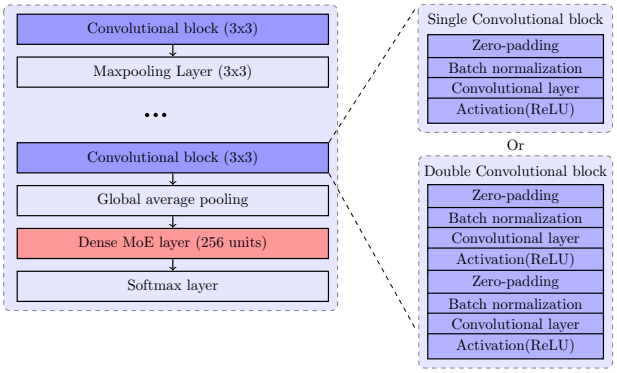

Fig. 3. A Mixture of Experts (MoE) layer embedded within a CNN model.

3.3. Convolutional Neural Networks

In this paper, we consider CNNs as extractors of high-level features and observe six different CNN structures by adjusting the depth of the CNNs and the structure of the convolutional blocks as well as different numbers of single convolutional blocks and double convolutional blocks which were used in the best model of the DCASE 2018 task 1B [16]. However, the purpose is to empirically determine only one suitable CNN structure that is able to learn informative high-level features for the MoE layer. In the DCASE 2018 task 1B [16], these CNN structures build component models and their outputs are fed to an ensemble model. Both approaches differ in the number of parameters.

Similarly, we proposed different CNN structures[2] that use either single convolutional blocks or double convolutional blocks as shown in Figure 3. A single convolutional block consists of zero-padding (1x1 size), batch normalization, 2D convolution layers (3x3 filter size) followed by Rectifier Linear Units (ReLUs) activation functions. A double convolutional block is a repeated structure of two single convolutional blocks. After the single/double convolutional block, we use a max-pooling layer (3x3 size) for the purpose of reducing dimensionality of the convolutional output and to ease the computation for the following layers as well as to reduce overfitting in the training phase. In addition, the pooling of the last convolutional block is replaced by global average pooling (GAP) layer follows the last convolution block instead of max-pooling. The GAP layer allows to reduce the number of outputs of the previous layer before feeding the data to the dense MoE layer. The aim is to maintain the global characteristics of each input sample, so that the ASC model is less bulky and sufficiently strong to deal with the complexity imposed by the mismatch in the data. Figure 3 shows the structure of the CNN model using single and double convolutional blocks.

[2]The models are implemented on Keras https://github.com/keras-team/keras

Based on the CNNs setup in [16] , we select the number of filters for the convolutional layers of the CNNs including 2, 3 and 4 single or double convolutional blocks as 32 - 256, 32 - 128 - 256 and 32 - 64 - 128 - 256, respectively. The same size of 3x3 filters is also selected for both convolutional layers of each double convolutional block.

3.4. Mixup data augmentation

Mixup [22] is an effective data augmentation method used in most of the best systems in the recent DCASE challenge [3], [4]. Mixup constructs virtual training examples by a convex combination of two randomly selected training data samples (x_i, y_i) and (x_j, y_j), i.e.

$$\begin{aligned} \tilde{x} &= \lambda x_i + (1-\lambda)x_j, \\ \tilde{y} &= \lambda y_i + (1-\lambda)y_j, \end{aligned} \tag{3}$$

where x_i and x_j are 2s samples (log-mel spectrogram) and y_i and y_j are one-hot encoded class labels i.e. output vectors. $\lambda \in [0, 1]$ is acquired by sampling from a beta distribution *Beta*(α, α) with α being a hyper parameter. We use an α of 0.2. We assess the impact of mixup data augmentation on the system performance.

4. EXPERIMENTS

4.1. Data

The audio data set for the ASC task 1B [1] is the TUT Urban Acoustic Scene 2018 Mobile data recorded in six European cities. It consists of 10 scenes. The development set is comprised of the task 1A data set recorded by using the same binaural microphone at a sampling rate of 48kHz. They are resampled and averaged into a single channel. A small amount of data is recorded by other devices. The original recordings were split into 10-second segments that are provided in the individual files. The data setup is as follows:

- Device A: 24 hours (8640 segments, 864 segments per acoustic scene)
- Device B: 2 hours (720 segments, 72 segments per acoustic scene)
- Device C: 2 hours (720 segments, 72 segments per acoustic scene)

The training subset is composed of 6122 segments from device A, 540 segments from device B, and 540 segments from device C. The test subset contains 2518 segments from device A, 180 segments from device B, and 180 segments from device C. Because the evaluation data set has been provided without ground truth, we use a training subset and a test subset of the development set for training and evaluating the models, respectively.

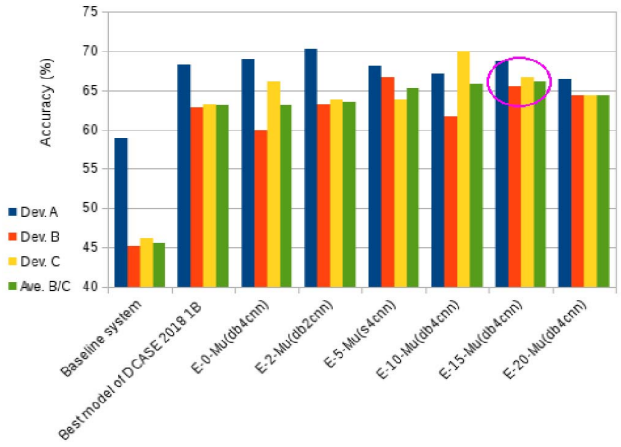

Fig. 4. Comparison of the baseline system, the best model of DCASE 2018 task 1B and the best MoE layer models with expert numbers ranging from 0 to 20.

4.2. Setup

The validation set accounts for approximately 30% of the original training data and there are no segments from the same location and city in both training and validation data sets.

Training the network is carried out by optimizing the categorical cross-entropy using the stochastic gradient decent optimizer at a learning rate of 0.01. We use Glorot uniform data to initialize the network weights. The number of epochs and batch size was 500 and 128, respectively. Data is shuffled between the epochs. Model performance is evaluated on the validation set after each epoch and the selected model is the best performing one on the validation set.

4.3. Performance on the test set

We run experiments for 6 CNN structures (db4cnn, db3cnn, db2cnn, s4cnn, s3cnn and s2cnn where db/s denote for double / single convolutional blocks) combined with mixup data augmentation. We use either no dense MoE layer or 2, 5, 10, 15 and 20 experts of the dense MoE layer. We represent the highest average performances for Device B and C as well as the performances among the three recording devices A, B and C. Furthermore, in order to assess the role of mixup data augmentation in our system, we test the vanilla CNN models and the best MoE layer configurations for the case of mixup/ no mixup. In addition, we test a model using mixup and convolutional MoE layers replacing the vanilla convolutional layers of our simplest CNN structure. The performances including the DCASE 2018 baseline system and the best model for task 1B of the DCASE 2018 challenge are presented in Table 1 and Figure 4.

Generally, we can see from Table 1 that the accuracy of our models for Device A is always higher than that of Device B and C, while the accuracy for Device B is almost always

lowest among the 3 devices. The reason could be the low quantity of the training set samples recorded by Device B and C compared to Device A. The quality of these devices could also affect the quality of the recordings where device B seems to be the worst device among them. In particular, it includes more noise as well as degradation of audio signal quality than the others. This causes more difficulties in scene classification.

Furthermore, the proposed models composed of 4 double convolutional blocks with or without mixup data augmentation outperform the baseline by approximately 15% in terms of accuracy. When using mixup the accuracy of *E-0-Mu(s3cnn)* is on par with the best model of the challenge for Device A while its site in term of parameters is approximately 30 times lower. Although there are some dense MoE layer models achieving 70% or higher accuracy for device A or C, they can not outperform the model using data mixup and 4 double convolutional blocks with 15 experts in the dense MoE layer i.e. *E-15-Mu(db4cnn)*. This is the best model among the proposed models. It achieves 66.1% classification accuracy. However, when keeping the same model structure and without the data mixup technique, the performance drops by 4.4% (absolute). Based on these results, we can conclude that mixup is the key factor leveraging the performance of our proposed models. Additionally, the 15 experts of convolutional MoE layer for two single convolutional blocks *Ecnn-15-Mu-s2cnn* can outperform the baseline system. However, this model comes with more than 35M parameters while there are around 2M or less than 2M parameters for the models using vanilla CNN layers and the dense MoE layer. The depth of the CNN model causes the large number of parameters.

Finally, our model outperforms the best DCASE 2018 system for task 1B in both accuracy i.e. 66.1% versus 63.6 % as well as model size i.e. around 2M versus 12M parameters, respectively. Table 2 shows class-wise accuracy of the baseline models of the DCASE 2018 challenge and our proposed model. By comparing performance of the individual scenes, we can see that the proposed model and the best model of the DCASE 2018 challenge have a similar accuracy for park and street-traffic. These are the easiest classes to recognize for both models while the easiest scene for the baseline system is bus. Metro is the most difficult scene for our model while the most difficult class to recognize for both compared models is tram.

5. CONCLUSION

This paper proposes a robust ASC system using a mixture of experts (MoE) layer as part of the CNN model. Furthermore, mixup data augmentation is used. Our system achieves 66.1% of accuracy on the test set of the DCASE 2018 task 1B. This is 2.5% (absolute) higher than of the best system in the same challenge task, while the complexity of our model in terms of number of parameters is approximately one sixth of the model used in the challenge. Furthermore, we evaluated different CNN structures as a high-level feature extractor for the MoE fully connected layer as well as the effect of the number of experts of the MoE layer. The model is able to deal with the mismatch of the recording devices in the DCASE 2018 challenge data. In addition, we show that the mixup data augmentation is really useful in our system leveraging the performance.

Table 1. Accuracy (in %) and number of parameters of corresponding models: Ecnn-15 denotes the model using a convolutional MoE layer with 15 experts, E-0 denotes a vanilla CNN without dense MoE layer, E-*i* denotes the model using dense MoE layer with *i* experts, Mu denotes mixup data augmentation, noMu denotes no mixup data augmentation, (db*j*cnn) denotes a CNN model including *j* double CNN blocks and s*j*cnn denotes a CNN model including *j* single CNN blocks.

Accuracy	Dev.A	Dev.B	Dev.C	Ave.(B,C)	Parameters
Baseline [1]	58.9 ($\pm$0.8)	45.1 $\pm$3.6	46.2 $\pm$4.2	45.6 $\pm$3.6	-
Best model of DCASE 2018 task 1B [16]	**68.4**	**63.3**	**63.9**	**63.6**	**12M**
E-0-noMu(db4cnn)	65.2	56.7	65.0	60.8	1,241,690
E-0-Mu(s3cnn)	69.0	60.0	66.1	63.1	401,610
E-2-Mu(db2cnn)	70.3	63.3	63.9	63.6	809,724
E-5-Mu(s4cnn)	68.1	66.7	63.9	65.3	721,451
E-5-Mu(db2cnn)	70.1	61.1	62.8	61.9	1,007,871
E-10-Mu(db4cnn)	67.1	61.7	70.0	65.8	1,836,388
E-15-Mu(db4cnn)	**68.7**	**65.6**	**66.7**	**66.1**	**2,166,633**
E-15-Mu(db2cnn)	70.8	59.4	60.0	59.7	1,668,361
E-15-noMu(db4cnn)	64.8	58.3	65.0	61.7	2,166,633
E-20-Mu(db4cnn)	66.4	64.4	64.4	64.4	2,496,878
E-20-Mu(s4cnn)	71.4	62.8	63.9	63.3	1,712,186
Ecnn-15-Mu-s2cnn	62.8	55.6	53.3	54.4	31,415,206

Table 2. Class-wise average accuracy of Device B and C of the *E-15-Mu(db4cnn)* system on the test set compared to the baseline system and the best model of DCASE 2018 task 1B [16].

Scene labels	Baseline [1]	Best model of DCASE 2018 task 1B [16]	Proposed
Airport	72.5	58.3	47.2
Bus	78.3	80.6	77.8
Metro	20.6	41.7	30.6
Metro station	32.8	61.1	75.0
Park	59.2	91.7	91.7
Public square	24.7	55.6	47.2
Shopping mall	61.1	75.0	83.3
Street_pedestrian	20.8	50.0	63.9
Street_traffic	66.4	83.3	83.3
Tram	19.7	38.9	61.1
Average	45.6 ± 3.6	63.6	**66.1**

6. ACKNOWLEDGMENT

This research was supported by the Vietnamese - Austrian Government scholarship and by the Austrian Science Fund (FWF) under the project number I2706-N31. We acknowledge NVIDIA for providing GPU computing resources.

7. REFERENCES

[1] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen, "A multi-device dataset for urban acoustic scene classification," in *Proceedings of DCASE 2018 Workshop*, November 2018, pp. 9–13.

[2] Yoonchang Han and Jeongsoo Park, "Convolutional neural networks with binaural representations and background subtraction for acoustic scene classification," in *Proceedings of DCASE 2017 Workshop*, November 2017, pp. 46–50.

[3] Yuma Sakashita and Masaki Aono, "Acoustic scene classification by ensemble of spectrograms based on adaptive temporal divisions," Tech. Rep., DCASE2018 Challenge, September 2018.

[4] Matthias Dorfer, Bernhard Lehner, Hamid Eghbalzadeh, Heindl Christop, Paischer Fabian, and Widmer Gerhard, "Acoustic scene classification with fully convolutional neural networks and I-vectors," Tech. Rep., DCASE2018 Challenge, September 2018.

[5] Tilak Purohit and Atul Agarwal, "Acoustic scene classification using deep CNN on raw-waveform," Tech. Rep., DCASE2018 Challenge, September 2018.

[6] Kun Qian, Zhao Ren, Vedhas Pandit, Zijiang Yang, Zixing Zhang, and Bjrn Schuller, "Wavelets revisited for the classification of acoustic scenes," in *Proceedings of DCASE 2017 Workshop*, November 2017, pp. 108–112.

[7] Shefali Waldekar and Goutam Saha, "Wavelet-based audio features for acoustic scene classification," Tech. Rep., DCASE2018 Challenge, September 2018.

[8] Yang Liping, Chen Xinxing, and Tao Lianjie, "Acoustic scene classification using multi-scale features," in *Proceedings of DCASE 2018 Workshop*, November 2018, pp. 29–33.

[9] Seongkyu Mun, Suwon Shon, Wooil Kim, David K Han, and Hanseok Ko, "A novel discriminative feature extraction for acoustic scene classification using rnn based source separation," *IEICE TRANSACTIONS on Information and Systems*, vol. 100, no. 12, pp. 3041–3044, 2017.

[10] Huy Phan, Philipp Koch, Fabrice Katzberg, Marco Maass, Radoslaw Mazur, and Alfred Mertins, "Audio scene classification with deep recurrent neural networks," in *Proceedings of Interspeech*, 2017.

[11] Y. Li, X. Li, Y. Zhang, W. Wang, M. Liu, and X. Feng, "Acoustic scene classification using deep audio feature and blstm network," in *2018 International Conference on Audio, Language and Image Processing (ICALIP)*, July 2018, pp. 371–374.

[12] Jinxi Guo, Ning Xu, Li-Jia Li, and Abeer Alwan, "Attention based cldnns for short-duration acoustic scene classification," in *Proceedings of Interspeech*, 2017, pp. 469–473.

[13] Xiang Long, Chuang Gan, Gerard de Melo, Xiao Liu, Yandong Li, Fu Li, and Shilei Wen, "Multimodal keyless attention fusion for video classification," AAAI, 2018.

[14] Zhao Ren, Qiuqiang Kong, Kun Qian, Mark Plumbley, and Bjrn Schuller, "Attention-based convolutional neural networks for acoustic scene classification," in *Proceedings of DCASE 2018 Workshop*, November 2018, pp. 39–43.

[15] Seongkyu Mun, Sangwook Park, David K Han, and Hanseok Ko, "Generative adversarial network based acoustic scene training set augmentation and selection using SVM hyper-plane," in *Proceedings of DCASE 2017 Workshop*, November 2017, pp. 93–102.

[16] Truc Nguyen and Franz Pernkopf, "Acoustic scene classification using a convolutional neural network ensemble and nearest neighbor filters," in *Proceedings of DCASE 2018 Workshop*, November 2018, pp. 34–38.

[17] D. Eigen, M. Ranzato, and I. Sutskever, "Learning factored representations in a deep mixture of experts," in *Proceedings of the International Workshop on Learning Representations (ICLR)*, 2014.

[18] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," in *Proceedings of the ICLR*, 2017.

[19] Zhilin Yang, Zihang Dai, Ruslan Salakhutdinov, and William W Cohen, "Breaking the softmax bottleneck: A high-rank rnn language model," in *Proceedings of the ICLR*, 2018.

[20] R. Jacobs, M. Jordan, S. Nowlan, and G. Hinton, "Adaptive mixtures of local experts," *Neural computation*, vol. 3, no. 1, pp. 79–87, 1991.

[21] Orhan Emin, "The softmax bottleneck is a special case of a more general phenomenon," 2018, https://severelytheoretical.wordpress.com.

[22] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proceedings of the ICLR*, 2018.