DCASE 2022: COMPARATIVE ANALYSIS OF CNNS FOR ACOUSTIC SCENE CLASSIFICATION UNDER LOW-COMPLEXITY CONSIDERATIONS

Technical Report

Josep Zaragoza-Paredes[1], *Javier Naranjo-Alcazar*[2], *Valery Naranjo*[1], *Pedro Zuccarello*[2]

[1] Universitat Politecnica de Valencia, ETSIT, Valencia, Spain, {jozapa@teleco, vnaranjo@dcom}.upv.es

[2] Instituto Tecnologico de Informatica, Valencia, Spain {jnaranjo, pzuccarello}@iti.es

ABSTRACT

Acoustic scene classification is an automatic listening problem that aims to assign an audio recording to a pre-defined scene based on its audio data. Over the years (and in past editions of the DCASE) this problem has often been solved with techniques known as ensembles (use of several machine learning models to combine their predictions in the inference phase). While these solutions can show performance in terms of accuracy, they can be very expensive in terms of computational capacity, making it impossible to deploy them in IoT devices. Due to the drift in this field of study, this task has two limitations in terms of model complexity. It should be noted that there is also the added complexity of mismatching devices (the audios provided are recorded by different sources of information). This technical report makes a comparative study of two different network architectures: conventional CNN and Conv-mixer. Although both networks exceed the baseline required by the competition, the conventional CNN shows a higher performance, exceeding the baseline by 8 percentage points. Solutions based on Conv-mixer architectures show worse performance although they are much lighter solutions.

Index Terms— DCASE, Acoustic Scene Classification, CNNs, TinyML, low-complexity

1. INTRODUCTION

Acoustic Scene Classification is understood as the task within the field of machine listening that aims at extracting context from audio data. This context is achieved by assigning a pre-defined scene to an audio clip [1, 2, 3, 4, 5]. Thus, the audio is referenced to the location where it is being recorded (park, airport, shopping mall, among others). Due to the characteristics of this problem, it can be defined as a supervised classification problem.

Since the first edition of the DCASE Challenge in 2013, this problem has always appeared as Task 1 [6, 7, 8]. However, it has not always been presented in the same manner as it has been modified over the years to reflect different constraints when deploying machine listening solutions, two of them being the appearance of the open set [9] or the limitation of mismatch devices [10]. Over the years, one of the most common practices by participants was the use of ensembles [11, 12, 13]. The idea behind this solution lies in the use of various machine learning models to improve the performance of the final system. By combining the output of several

systems, a more robust prediction is obtained [2]. However, these solutions are limited by the number of operations that need to be performed to obtain a final output. This makes it impossible (or challenging) to deploy these solutions in IoT devices.

This year's edition (2022) presents two considerations for the system to be taken as valid:

- The maximum number of parameters is set to 128K and the used variable type is fixed into INT8
- The maximum number of multiply-accumulate operations is set to 30M

With this in considerations, it has been decided to design two solutions:

- Solution based on a convolutional module composed of a conventional convolutional layer and a separate convolutional layer (Conv-Sep)
- Solution based on the module known as the Conv-mixer

Both solutions have the same structure (number of filters, pool sizes, dropouts, etc.) so that they can be fully comparable (for more information see 2.2). To meet the requirements of the task, the models have been converted to a TFLite format with 8-bit quantization.

In terms of the performance of the proposed systems, both solutions exceed the baseline proposed by the organisation. While the Conv-Sep solution shows better performance with the same number of filters, it also shows a higher number of MMACs and parameters. All these results are discussed in Section 3.

The rest of the technical report is divided as follows: section 2 presents the different blocks of the system (audio representation, machine learning model and training process). Section 3 shows the results. Finally, section 4 concludes our work.

2. METHOD

The following subsections explain the modules that make up the system presented to the Challenge.

2.1. Audio representation

The first module of the system consists of obtaining a time-frequency representation of the audio, $F \times T$, where F is the number of frequency bins and T is the time frames. For temporal anal-

Model architecture
Log-Mel spectrogram representation $64 \times 51 \times 1$
Convolutional module (Number of filters)
MaxPooling(1,4)
Dropout(0.3)
Convolutional module (Number of filters)
MaxPooling(1,2)
Dropout(0.3)
Global Average Pooling
Dense(10)

Table 1: Base architecture of the model used in the Challenge. The Convolutional module corresponds to the module explained in the different sections of subsection 2.2. The different values in parenthesis correspond to the hyperparameters used. In the MaxPooling layer the pool size is defined, in the Dropout layer the rate is specified and in the Convolutional module the number of filters. The Dense layer has 10 units, corresponding to the number of classes to predict

ysis, a window of 40ms with 50% overlap is used. The sample rate is set to 44100 Hz. For frequency resolution, it has been decided to use a bank of 64 Mel filters. Finally, the logarithm is performed to obtain the log-Mel spectrogram [14, 15]. This configuration crates a 64×51 audio log-Mel spectrogram from an audio of 1 second duration. This module has been implemented using the Librosa Python package.

2.2. System description

The model used has a VGG-style architecture [16]. Convolutional layers are interspersed with pooling and dropout layers. In this paper, two convolutional layer structures are compared. The first one is based on a two-layer convolutional module and the second one is based on the Conv-mixer module. The base model architecture can be seen in Table 1.

2.2.1. Conv-Sep model

This architecture is based on the VGG architecture. Two convolutional layers are stacked followed by pooling layers (dimensionality reduction) and dropout layers (avoiding overfitting). However, a number of modifications have been made with respect to a conventional VGG network. ReLU activation layers are replaced by ELU layers. To meet the complexity requirements imposed in the Challenge, the second convolutional layer is replaced by a Separable convolution layer. Thus, the convolutional module can be seen in Table 2

2.2.2. Conv-mixer model

The Conv-mixer module was presented in [17]. It consists of a module with two convolutional layers, the first one being DepthWise and the second one PointWise. The two layers are followed by a GELU activation layer and a BatchNormalization layer. In addition, there

Convolutional module Conv-Sep
Conv Layer (number of filters)
Batch Normalization
ELU
Separable Conv Layer (number of filters)
Batch Normalization
ELU

Table 2: Architecture of the convolutional module of the Conv-Sep model. The number of filters in the two convolutional layers is the same

is a residual connection between the previous point of the first layer and the result obtained from this layer. Prior to the first convolutional module, path embedding is performed. The path size is set to 1. For more details, please see [17].

2.3. Experimental details

2.3.1. Training details

The following configuration has been used in the training phase:

- The maximum number of epochs is set to 500
- The training is stopped if accuracy is not improved within 30 epochs
- The learning rate is reduced by a factor of 0.5 if the accuracy is not improved within 15 epochs
- The optimizer used is Adam
- Loss function is set to categorical crossentropy

All models have been implemented with the ML library Tensorflow.

2.3.2. Dataset

The dataset used in this task is the TAU Urban Acoustic Scenes 2022 Mobile, development dataset. The dataset is composed of recordings from 4 different real devices and 11 simulated devices. The recordings have been made in 12 cities, 10 of which are present in the development set. The audios have a duration of 1 second. The training set is composed of 40 hours of audio provided in mono, 44100Hz sampling rate and 24-bit format. The classes present are 10 (as in the other editions). In summary, the training set consists of 139970 segments and the test set consists of 29680 segments.

2.4. Model quantization

Following the requirements of the task. The weights of the model have undergone a Post-training quantization. For this procedure, the tools present in Tensorflow have been used to convert and quantise models to the TFLite format.

3. RESULTS

The results obtained can be seen in Table 3. The models that have been submitted to the Challenge are shown in green.

Model	Number of filters	Accuracy(%)	Log Loss	MMACs	Number of parameters
Challenge baseline	[16-32]	42.90	1.58	29,234,920	46,512
Conv-Sep	[40-40]	50.00	**1.44**	**20,306,320**	**20,088**
Conv-mixer	[40-40]	46.32	1.50	17,979,280	10,515
Conv-Sep	[48-48]	**50.57**	**1.44**	28,570,080	28,320
Conv-mixer*	[48-48]	47.26	1.64	24,671,712	14,139
Conv-Sep	[32-64]	50.09	1.45	23,138,944	26,544
Conv-mixer	[32-64]	46.76	1.64	15,895,168	12,683
Conv-Sep	[64-64]	51.12	1.49	49,300,096	49,008
Conv-mixer	[64-64]	47.79	1.70	41,153,152	22,923

Table 3: Results obtained using proposed network with convolutional layer configuration and conv-mixer before quantization. The model indicated as * was the one submitted to the Challenge in TFLite format and int8 quantization

An analysis of each filter configuration has been carried out for each of the models. As can be seen, the number of parameters and MMACs is drastically lower with the Conv-mixer configuration. However, the most promising results are obtained for the model with the Conv-Sep module.

As can be seen, the configuration of two Conv-Sep modules with 48 filters per module. The accuracy improves the baseline of the Challenge with about 8 perceptual points and the Log Loss is reduced with a value of 0.14.

Solutions based on Conv-mixer modules show a worse performance compared to Conv-Sep. The Conv-mixer based solution presents a great improvement in terms of complexity, being these solutions very lightweight. However, the best solution only improves by 5% the system. This configuration sometimes improves the accuracy but worsens the Log Loss.

4. CONCLUSION

The results obtained in this submission show that the performance of an ASC system can be improved while reducing its complexity. Two architectures based on convolutional layers are analysed in this paper. The Conv-mixer based solutions show an improvement in complexity. However, no configuration has been found to improve the Conv-Sep setup. Solutions based on VGG architectures (interleaving convolutional and separable layers) show a great performance-complexity trade-off.

5. ACKNOWLEDGMENT

6. REFERENCES

[1] J. Naranjo-Alcazar, S. Perez-Castanos, P. Zuccarello, and M. Cobos, "Acoustic scene classification with squeeze-excitation residual networks," *IEEE Access*, vol. 8, pp. 112 287–112 296, 2020.

[2] S. Perez-Castanos, J. Naranjo-Alcazar, P. Zuccarello, M. Cobos, and F. J. Ferri, "Cnn depth analysis with different channel inputs for acoustic scene classification," *arXiv preprint arXiv:1906.04591*, 2019.

[3] D. Battaglino, L. Lepauloux, N. Evans, F. Mougins, and F. Biot, "Acoustic scene classification using convolutional neural networks," *IEEE AASP Challenge on Detec*, 2016.

[4] A. Singh and M. D. Plumbley, "A passive similarity based cnn filter pruning for efficient acoustic scene classification," *arXiv preprint arXiv:2203.15751*, 2022.

[5] L. Pham, K. Dinh, D. Ngo, H. Tang, and A. Schindler, "Wider or deeper neural network architecture for acoustic scene classification with mismatched recording devices," *arXiv preprint arXiv:2203.12314*, 2022.

[6] D. Giannoulis, D. Stowell, E. Benetos, M. Rossignol, M. Lagrange, and M. D. Plumbley, "A database and challenge for acoustic scene classification and event detection," in *21st European Signal Processing Conference (EUSIPCO 2013)*, Sep. 2013, pp. 1–5.

[7] A. Mesaros, T. Heittola, and T. Virtanen, "TUT database for acoustic scene classification and sound event detection," in *24th European Signal Processing Conference 2016 (EUSIPCO 2016)*, Budapest, Hungary, 2016.

[8] A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj, and T. Virtanen, "DCASE 2017 challenge setup: Tasks, datasets and baseline system," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017)*, November 2017, pp. 85–92.

[9] H. Zhu, C. Ren, J. Wang, S. Li, L. Wang, and L. Yang, "DCASE 2019 challenge task1 technical report," DCASE2019 Challenge, Tech. Rep., June 2019.

[10] A. Mesaros, T. Heittola, and T. Virtanen, "A multi-device dataset for urban acoustic scene classification," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, November 2018, pp. 9–13. [Online]. Available: https://arxiv.org/abs/1807.09840

[11] J. Huang, P. Lopez Meyer, H. Lu, H. Cordourier Maruri, and J. Del Hoyo, "Acoustic scene classification using deep learning-based ensemble averaging," DCASE2019 Challenge, Tech. Rep., June 2019.

[12] B. Ding, G. Liu, and J. Liang, "Acoustic scene classification based on ensemble system," DCASE2019 Challenge, Tech. Rep., June 2019.

[13] Y. Haocong, S. Chuang, and L. Huiyong, "Acoustic scene classification using CNN ensembles and primary ambient extraction," DCASE2019 Challenge, Tech. Rep., June 2019.

[14] B. Kim, S. Yang, J. Kim, and S. Chang, "QTI submission to DCASE 2021: Residual normalization for device-imbalanced acoustic scene classification with efficient design," DCASE2021 Challenge, Tech. Rep., June 2021.

[15] C.-H. H. Yang, H. Hu, S. M. Siniscalchi, Q. Wang, W. Yuyang, X. Xia, Y. Zhao, Y. Wu, Y. Wang, J. Du, and C.-H. Lee, "A lottery ticket hypothesis framework for low-complexity device-robust neural acoustic scene classification," DCASE2021 Challenge, Tech. Rep., June 2021.

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

[17] A. Trockman and J. Z. Kolter, "Patches are all you need?" *arXiv preprint arXiv:2201.09792*, 2022.