AI4EDGE SUBMISSION TO DCASE 2023 LOW COMPLEXITY ACOUSTIC SCENE CLASSIFICATION TASK1

Technical Report

Carlos Almeida[1], *Federico Piovesan*[2], *Luis Conde Bento*[1,3], *Mónica Figueiredo*[1,4],

[1] Polytechnic of Leiria, Leiria, Portugal
[2] Polytechnic of Torino, Torino, Italy
[3] Institute of Systems and Robotics, Coimbra, Portugal
[4] Institute of Telecommunications, Portugal

ABSTRACT

The DCASE task 1 challenge [1] aims to classify acoustic scenes using devices with low computational power and memory. The DCASE2023 challenge gives further importance to the size and multiply-accumulate operation count (MAC), this report aims to describe the submission to this challenge, following our research group's previous work in this field, and the model submitted to DCASE 2022 [2]. We use a one-versus-all ten-network ensemble model and propose a knowledge distillation custom method to reduce model complexity. The ensemble model is used as the *teacher* network, distilling knowledge to the *student*. The student has 3 variations, the first model is a tuned version of the DCASE2022 baseline architecture, for the second model a slightly larger version of the first model and for the third model a larger version of the second model using structured pruning to further reduce model complexity. Data preprocessing is also conducted in order to further improve performance. Results show that the proposed knowledge distillation methods were able to improve the accuracy significantly.

***Index Terms*—** DCASE2023, ensemble, knowledge distillation, data pre-processing, pruning

1. INTRODUCTION

Classification of acoustic scenes aims to identify different scenarios based on the audio features present on recorded audio. Many applications use information of the scenes where audio was recorded, such as automatic audio surveillance, robotics sensing, multimedia content analysis and machine listening [3], many of these applications have memory and computational restraints.

The task at hand, Low-Complexity Acoustic Scene Classification of the DCASE2023 Challenge has the goal of promoting the research around acoustic scene classification by comparing different approaches. The dataset used on this task is the TAU Urban Acoustic Scenes 2022 Mobile Dataset [4], this dataset includes audio data recorded from 9 devices either real or simulated. This challenge imposes system complexity limitations, these limitations are a maximum of 128 kilobyte (kB) including zero-valued parameters. Another limitation is also imposed, a maximum of 30 million MACs, these requirements were based on the constraints of Cortex-M4 devices. To improve the baseline model, with an accuracy of 42.9%, several techniques were studied and implemented, namely data preprocessing, model architecture tuning, knowledge distillation and structured pruning.

The main contribution of this work is on a new hybrid integration of multiple knowledge distillation techniques for acoustic scene classification. This report is organized as follows: Section 2 focuses over the data preprocessing conducted on the initial dataset aiming to improve accuracy; Section 3 discusses the submitted models architecture; Section 4 discusses the knowledge distillation methods used to improve the models performance; Section 5 details the structured pruning used to reduce the MAC usage; Section 6 presents and discusses the results of the submitted models; Section 7 presents the findings of this paper and concludes this report.

2. DATA PREPROCESSING TECHNIQUES

Data preprocessing techniques have proven effective at facilitating the extraction of more relevant features in audio data. The dataset used on this task is the TAU Urban Acoustic Scenes 2022 Mobile Dataset [4], this dataset includes audio data recorded from 9 devices either real or simulated.

The original data is provided in a single-channel 44.1kHz 24-bit format, however down-sampling the data proved to greatly diminish the amount of data, and consequently the training time with a minor loss in relevant information. The Librosa library [5] was used to downsample the input data to 8kHz, additionally the log Mel spectogram of the downsampled audio data was calculated using Short-Time Fourier Transform (STFT) with a windoww length of 2048 samples.

Data augmentation techniques such as pitch shift, time stretch, mixup [6], time and frequency masks [7] were also used to balance the difference of samples available between devices and improve the model's generalization.

Additionally, building upon our team earlier work [8], the Kapre [9] tool with KerasTuner[10] and Hyperband [11] method was used to search for the optimal hyperparameters for signal representations that best contribute to a higher accuracy, the configuration that presented the best results and therefore is used on this submission had a sampling frequency of 8kHz with 140 Mel frequency bands.

3. ARCHITECTURE

3.1. Teacher and student architectures

The teacher and student architectures were based on a previous submission to the DCASE challenge [8]. The student is based on a convolutional neural network, provided by the DCASE competition organizers, which had an accuracy of 42.9% on the development dataset with 46.5k parameters and 29.23 million MACs. The student model was then hyper-tuned to find the best performing architecture within a set of constraints imposed by the competition.

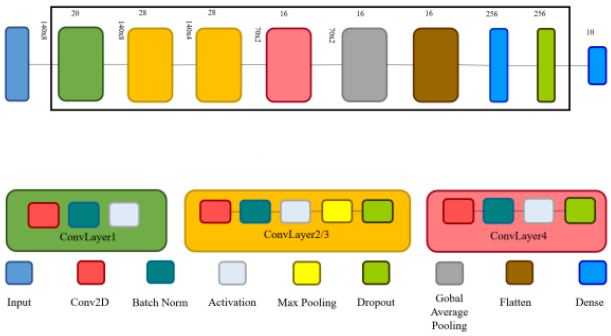

Figure 1: Student architecture [8].

The model is represented in Figure 1 is composed of 4 layers, which can encompass convolutional layers, dropout layers, to reduce overfitting; and max/average pooling layers, to reduce the size of the feature maps and consequently the number of parameters of the neural network. There are 3 slight variations to the student model, the **student model 1 (SM1)** has a total of 52852 parameters, 25 million MAC and its basis, the convolutional layers have the following parameters:

- CNN layer #1 - 20 filters with a kernel size of 7x5
- CNN layer #2 - 28 filters with a kernel size of 7x3
- CNN layer #3 - 28 filters with a kernel size of 3x7
- CNN layer #4 - 16 filters with a kernel size of 7x5

The second variation, the **student model 2 (SM2)** has 2 more convolutional layers and a total of 68996 parameters, 29.3 million MAC, its convolutional layers have the following parameters:

- CNN layer #1 - 20 filters with a kernel size of 7x5
- CNN layer #2 - 24 filters with a kernel size of 7x3
- CNN layer #3 - 24 filters with a kernel size of 7x3
- CNN layer #4 - 24 filters with a kernel size of 3x7
- CNN layer #5 - 24 filters with a kernel size of 3x7
- CNN layer #6 - 16 filters with a kernel size of 7x5

The third and final variation, the **student model 3 (SM3)** has a total of 86116 parameters, 37 million MAC and its convolutional layers have the following parameters:

- CNN layer #1 - 20 filters with a kernel size of 7x5
- CNN layer #2 - 28 filters with a kernel size of 7x3
- CNN layer #3 - 28 filters with a kernel size of 7x3
- CNN layer #4 - 28 filters with a kernel size of 3x7
- CNN layer #5 - 28 filters with a kernel size of 3x7
- CNN layer #6 - 16 filters with a kernel size of 7x5

It's worth noting that the SM3 does not fit the limitations imposed by the DCASE challenge, this was done purposefully as this model will employ structured pruning to reduce its computational and memory requirements.

The teacher is a one-versus-all ten-model ensemble network, composed of 10 SM1 models, as represented in Figure 2. Each one of the models is trained to recognise one of 10 acoustic scenes. The teacher model has a total of 519770 parameters, about 10 times more than the SM1 model, which enables it to learn more complex features.

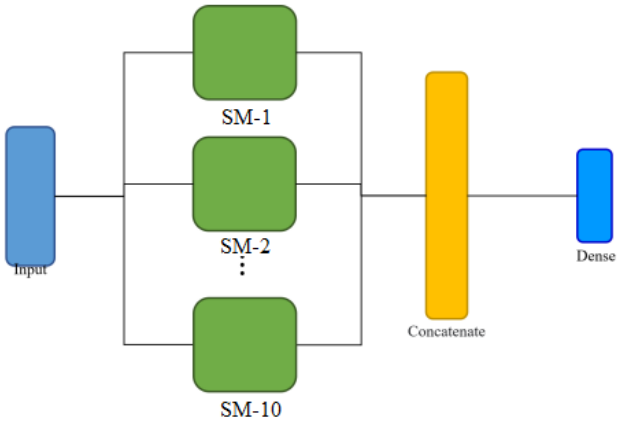

Figure 2: Teacher architecture [8].

4. KNOWLEDGE DISTILLATION METHOD

Relational Response Stagewise (RRS) knowledge distillation is the proposed mix of state-of-art knowledge distillation techniques. This method implements a combination of three techniques, relational-based knowledge distillation[12], response-based knowledge distillation[13] and the stage-wise training of Fitnets[14].

4.1. Relation-Based Knowledge Distillation

This method was first proposed in [12]. The goal is to transfer structural knowledge in the teacher's output, to the student, as is represented in Figure 3. Where $(x_1...x_n)$ represent data examples about to go into the teacher and student model, $(t_1...t_n)$ represents the teacher's representation of that data, and the student's representation is $(s_1...s_n)$. The relational potential, also known as distance-wise potential and angle-wise potential, of each representation is $\psi(t_1...t_n)$.

The distillation loss is based on the difference between the relational potential (ψ) of the teacher's output and the relational potential on the student's output, given by *distance-wise potential* and the *angle-wise potential*. Whereas the *distance-wise potential* is a measure of the Euclidean distance between pairs of data examples in the output representation space of the teacher and student models, and the *angle-wise potential* is the measure of the angle formed by triplets of data examples in the output representation space of the teacher and student models.

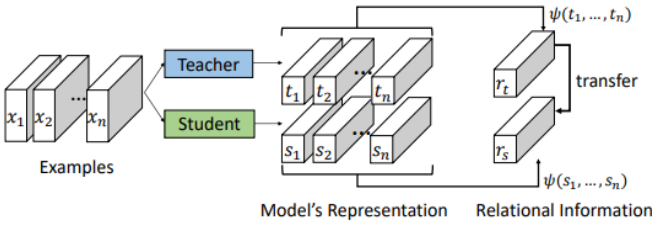

Figure 3: Relational-Based knowledge distillation [12].

The distillation loss is the difference in the angle and distance between the outputs of both models, as illustrated in Figure 4, where $(t_1...t_n)$ refers to the teacher's representation of data and $(s_1...s_n)$ refers to the student's.

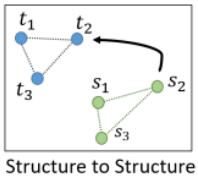

Figure 4: Structure of outputs of teacher and student [12].

One advantage of using this method is that it allows for more flexibility in transferring knowledge between models with different architectures or output dimensions. By focusing on preserving the structural relationships between data examples, rather than matching individual outputs.

4.2. Response-Based Knowledge Distillation

The goal of this method is provide the student the ability to mimic the predictions made by the teacher, using outputs of the last layer of the models - the probabilities that the input belongs to each of the classes (soft-targets or logits) [13].

To achieve the prior, a new loss function, called the distillation loss, captures the difference between both the outputs of the student and the teacher models. This distillation loss is minimized over training, so the student will learn to do the same predictions as the teacher. This process is represented on Figure 5.

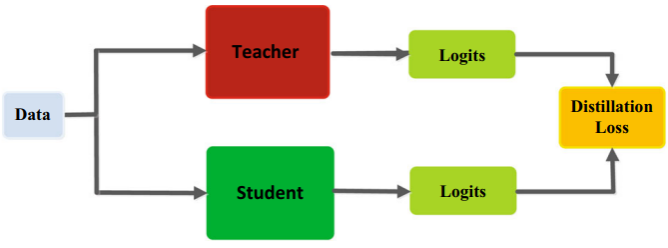

Figure 5: Response-Based knowledge distillation [15].

4.3. Relational Response Stagewise (RRS)

RRS knowledge distillation is here proposed to combine a set of state-of-art knowledge distillation techniques. This method implements a combination of three techniques: relational-based knowledge distillation; response-based knowledge distillation; and the stage-wise training of Fitnets.

RRS knowledge distillation includes a two-stage training procedure. The first stage consists of using the relation between outputs of the teacher and the student using the method described in Section 4.1.

The second stage of training compares the outputs of both the teacher and the student model and its difference is the distillation loss. This is based on response-based knowledge distillation described on Section 4.2.

This two-stage training allows the student to focus its training on one knowledge at a time, the first stage enables the student to learn to mimic the relations between outputs that it should give according to the teacher. Where the second stage focuses on teaching the student to mimic the outputs themselves, having already learned the relations that the outputs should have between each other on the first stage. By doing so, this two-stage training method allows the student to learn to mimic the outputs of the teacher more accurately.

5. STRUCTURED PRUNING

Structured pruning is a method used to reduce the computational cost and memory load, on a convolutional neural network (CNN) this can be achieve by removing entire filters, channels or layers from a network model. This can accelerate the inference process and facilitate their applications on memory constrained devices. The overall goal is to remove the least important structures from the network while minimizing the impact on accuracy.

A common method structured pruning method on CNN is to identify the least important filters, that have less impact on the performance, by their L1-norm [16] and remove them. The L1-norm of a filter w with n weights is calculated as:

$$\|w\|_1 = \sum_{i=1}^{n} |w_i| \tag{1}$$

This structured pruning method is implemented on model SM3 to reduce its complexity to fit the challenge requirements. A total of 20 "less relevant" filters are removed leading to the following change in the SM3s architecture:

- CNN layer #1 - 20 filters with a kernel size of 7x5
- CNN layer #2 - 22 (from 28) filters with a kernel size of 7x3
- CNN layer #3 - 22 (from 28) filters with a kernel size of 7x3
- CNN layer #4 - 24 (from 28) filters with a kernel size of 3x7
- CNN layer #5 - 24 (from 28) filters with a kernel size of 3x7
- CNN layer #6 - 16 filters with a kernel size of 7x5

This significant reduction leads to the following changes : **number of parameters** were reduced from 86.1k to 61.2k (71% of original size); **multiply-accumulate operation count** from 37M to 25.7M (69% of original count). These reductions were made to fit the requirements of the competition.

6. RESULTS AND SUBMISSIONS

The submitted models were trained for 200 epochs using a batch size of 64. An EarlyStopping callback was implemented to stop the training if progress on the validation accuracy was not being achieved, the models were also quantized to INT8.

The three submitted models (SM1,SM2, SM3) were obtained using the RRS knowledge distillation method. additionally upon SM3 structured pruning was also applied (described on Section 5).

Table 1 presents the results of the submitted models in terms of their accuracy on the validation dataset. The additional parameters namely the model size (number of parameters), peak memory usage and MACs (all computed using NeSsi [17]), were obtained while inference was perform on the evaluation dataset.

Table 1: Submitted models results

Model	Accuracy	# Parameters	Peak memory	MACs
SM1	67.46%	52.85k	62.7 kB	25.47 M
SM2	69.92%	68.99k	53.8 kB	29.30 M
SM3	66.05%	65.19k	49.3 kB	26.71 M

The smallest model parameter wise (SM1) has the highest peak memory usage. This is due to having the layers with the most filters (28). As acknowledged on the previous submission [8], to avoid overfitting, all models were trained solely with the train dataset.

7. CONCLUSIONS

This work was developed with the goal of studying the effectiveness of knowledge distillation in improving smaller and less complex neural networks for acoustic scene classification. The base system provided by DCASE2023 challenge had an accuracy of 42.9% with 46.5k parameters and 29.23 million MACs, after data preprocessing, model tuning and knowledge distillation an accuracy of 67.46% was achieved for a model with a similar number of parameters (SM1) while lowering the number of MACs. All submitted models increased the accuracy significantly, this showcases the flexibility and generalization capabilities of the proposed RRS knowledge distillation method.

8. REFERENCES

[1] https://dcase.community/challenge2023/task-low-complexity-acoustic-scene-classification.

[2] R. Anastácio, L. Ferreira, F. Mónica, and C. B. Luís, "Ai4edgept submission to DCASE 2022 low complexity acoustic scene classification task1," DCASE2022 Challenge, Tech. Rep., June 2022.

[3] J. Xie and M. Zhu, "Investigation of acoustic and visual features for acoustic scene classification," *Expert Systems with Applications*, vol. 126, pp. 20–29, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417419300661

[4] T. Heittola, A. Mesaros, and T. Virtanen, "TAU urban acoustic scenes 2022 mobile, development dataset," 2022.

[5] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, vol. 8, 2015.

[6] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," 2018.

[7] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Interspeech 2019*. ISCA, sep 2019. [Online]. Available: https://doi.org/10.21437%2Finterspeech.2019-2680

[8] R. Anastácio, L. Ferreira, F. Mónica, and C. B. Luís, "Ai4edgept submission to DCASE 2022 low complexity acoustic scene classification task1," DCASE2022 Challenge, Tech. Rep., June 2022.

[9] K. Choi, D. Joo, and J. Kim, "Kapre: On-gpu audio preprocessing layers for a quick implementation of deep neural network models with keras," 2017.

[10] T. OMalley, E. Bursztein, J. Long, F. Chollet, and L. I. H. Jin, "Kerastuner," https://github.com/keras-team/keras-tuner, 2019.

[11] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," 2018.

[12] W. Park, D. Kim, Y. Lu, and M. Cho, "Relational knowledge distillation," 2019.

[13] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015.

[14] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," 2015.

[15] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *CoRR*, vol. abs/2006.05525, 2020. [Online]. Available: https://arxiv.org/abs/2006.05525

[16] A. Zbiciak and T. Markiewicz, "A new extraordinary means of appeal in the polish criminal procedure: the basic principles of a fair trial and a complaint against a cassatory judgment," *Access to Justice in Eastern Europe*, vol. 6, no. 2, pp. 1–18, Mar. 2023.

[17] A. Ancilotto, "Nessi," https://github.com/AlbertoAncilotto/NeSsi, 2023.