END-TO-END CNN OPTIMIZATION FOR LOW-COMPLEXITY ACOUSTIC SCENE CLASSIFICATION IN THE DCASE 2021 CHALLENGE

Technical Report

Carlos A. Galindo-Meza[1]*, Juan A. del Hoyo Ontiveros*[2]*, Jose I. Torres Ortega*[2]*, Paulo Lopez-Meyer*[2]

[1] ITESO, Anillo Perif. Sur Manuel Gómez Morín 8585, Tlaquepaque, JAL,45604, Mexico, ms729621@iteso.mx

[2] Intel GDC, Intel Labs, Av. Del Bosque 1001, Zapopan, JAL, 45019, Mexico, {juan.antonio.del.hoyo.ontiveros, jose.torres.ortega, paulo.lopez.meyer}@intel.com

ABSTRACT

For the DCASE 2021 challenge we implemented an optimization pipeline to comply with the low-complexity restrictions specified with the Task 1a constraints. Initially, we trained and validated an end-to-end convolutional neural networks-based audio classification model following a typical deep learning training strategy. We then applied an efficient pruning procedure based on the lottery ticket hypothesis, and finally we executed a training-aware quantization to convert the model's weights from FP32 to INT8 format. Experimentation proved the feasibility of this approach by obtaining accuracy results above the baseline models reported in the challenge guidelines.

Index Terms— Acoustic Scene Classification, Low-Memory, Low-complexity, Deep Learning, End-to-End Audio Classification, Pruning, Quantization.

1. INTRODUCTION

For the 2021 Detection and Classification of Acoustic Scenes and Events challenge (DCASE2021), acoustic data were provided to solve different acoustic related tasks. Task 1a refers to the challenge of building a model to classify different recordings into predefined classes corresponding to different urban environment scenes [1].

This challenge's dataset consists of 10-second audio recordings obtained in 10 different acoustic scenes from 12 major European cities, grouped in three major classes: airport, bus, metro, metro station, park, public square, shopping mall, street pedestrian, street traffic, and tram [2]. This acoustic dataset comprises audio signals at 44.1 kHz of sampling rate in 24 bit resolution.

The challenge suggests the usage of a 1-fold arrangement for development as part of this task, with 70% for training and 30% for testing. Through the development stage of our implementations, we used Google Audioset data [3] to construct an efficient audio embedding generator.

2. METHODOLOGY

Following the guidelines provided by the challenge in the Task1 subtask a (Task 1a), we experimented with one low-memory implementation pipeline of an audio classification convolutional neural network architecture (CNN) through two optimization techniques: pruning of models using the lottery ticket hypothesis approach, followed by a FP32 to INT8 quantization. An end-to-end (e2e) CNN architectures was used as the base model and subject to optimization; this CNN takes raw audio data as the input into two 1D convolutional layers followed by a 2D multi-layer CNN. Pytorch was the framework of choice for our experimental setups.

In the following subsections, we describe in detail the experimentation followed around our low-memory implementation that constitutes our submissions to the DCASE2021 Task 1a challenge.

2.1. E2E CNN baseline model

The baseline e2e CNN architecture takes raw time-domain input waveform, as opposed to more commonly used spectral features, e.g. Log-Mel filterbank or Mel-frequency cepstral coefficients. The motivation for the development of these types of e2e architectures is that the front-end feature makes no assumptions of the frequency response, i.e. its feature representation is learned in a data-driven manner, thus are optimized for the task at hand provided there are sufficient training data.

For this implementation, we based our topology on the settings corresponding to the AemNet audio embeddings generator work described in [4], using a width multiplier of 0.5, and conventional depth-wise convolution layers. AemNet was pre-trained with Audioset [4, 3] to generate a vector of 512 audio embeddings that are sent to a fully-connected layer classifier built with ReLU activation functions in a transfer learning manner. Raw audio data from the Task 1a dataset was downsampled to 16 kHz and fed to the pre-trained e2e CNN, where the generated embeddings were used to train the classifier. Also during training, we allowed the weights in AemNet to fine-tune for the DCASE data over the backpropagation step.

We performed a search for the optimal parameters of this e2e acoustic classification CNN model. We experimented with different values and configurations that yield the best performing models, e.g. learning rate of 1×10^{-4} for the classifier layer and 1×10^{-5} for the AemNet embeddings generator, learning rate decay of 1×10^{-2}, weight decay of 1×10^{-4}, and drop out rate of 0.2. Additionally, in order to increase the robustness of the training process, we also used different audio data augmentation techniques commonly used in audio processing, such as random noise addition, random cropping of 1-second of the audio signal, and random gain variation, together with the widely used mixup data augmentation technique [5]. During the training, acoustic data were randomly selected to form mini-batches of training clips. At testing time, we run the inference over each complete audio file.

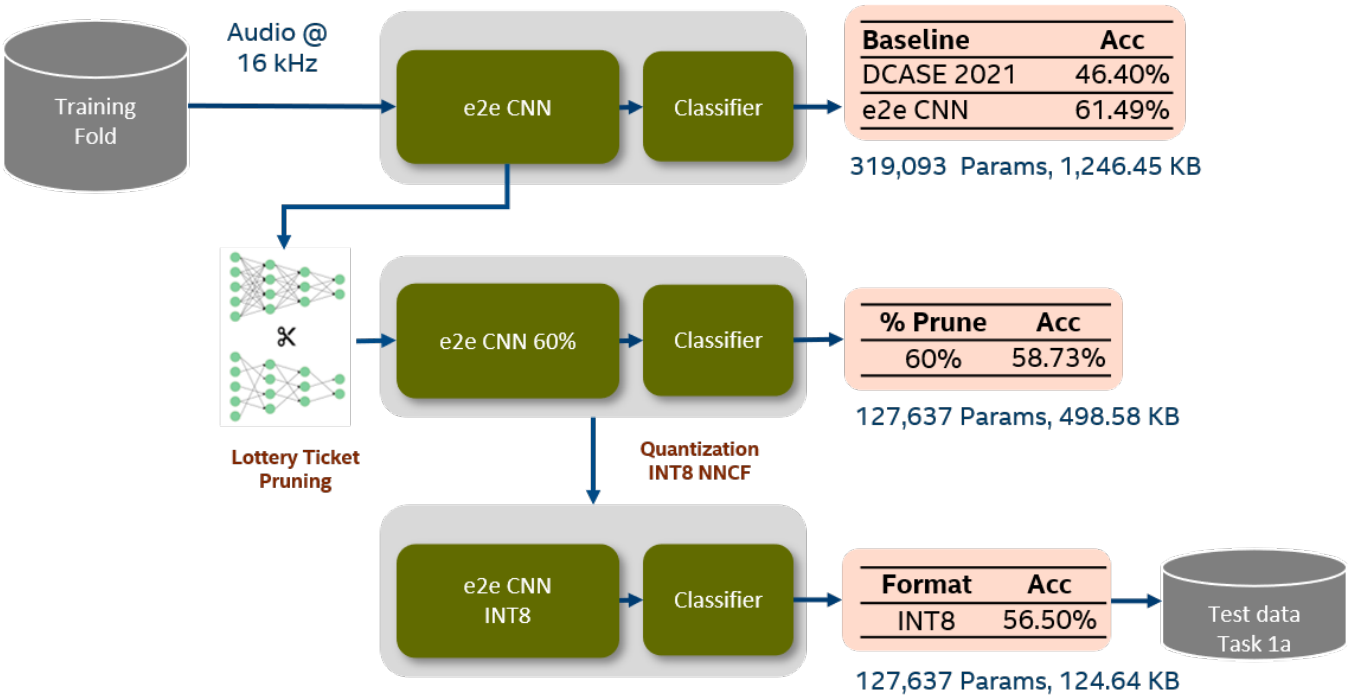

Figure 1: Development of our optimization pipeline for low-memory e2e CNN architecture for acoustic scene classification in the DCASE2021 Task 1a challenge.

2.2. Pruning based on the lottery ticket hypothesis

The resulting e2e CNN base model described above, with 319,093 parameters, is pruned at 60% in order to have a final model of 127,637 parameters. This pruning was executed through the lottery ticket hypothesis [6]. We initially trained our model to generate our base e2e CNN; after training, we removed 60% of the parameters by a typical pruning scheme, i.e. remove the parameters that are contributing less to the model's classification behavior. The lottery ticket hypothesis comes into place when, after identifying the post-pruning weights, a new training process is carried out with the original randomly initialized weights values assigned at the initial pre-training stage. This constitutes the spirit of the lottery ticket proposal, where subnetworks can be found in post-training pruning, that could reach testing accuracy comparable to the original network.

2.3. INT8 quantization

The resulting pruned e2e CNN constitutes an FP32 base model with 127,637 trainable parameters, which yields into 498.58 KB of memory size, clearly above the 128 KB restriction in the challenge. In order to decrease the memory size of this model, we applied a straight FP32-to-INT8 training-aware quantization based on the methodology described in [7], through the use of the available tool accessible in [8], that results in an optimized 124.64 KB e2e audio classification CNN model.

3. RESULTS AND DISCUSSION

The experimental results obtained by our optimization pipeline are displayed in the Tables 1 and 2. Table 1 shows the performance of the initially obtained e2e CNN in FP32 format over the Task 1a testing dataset. This e2e CNN constitutes the base model for the the lottery ticket prunig (e2e CNN LT) and subsequently for the INT8 training-aware quantization (e2e CNN INT8). It is not surprising to see a higher accuracy performance of the base model as compared to the LT and INT8 implementations. It can also be observed that the optimized model achieves a higher performance than the baseline reported in the Task 1a guidelines, with 56.50% vs 46.40% of acoustic scene classification, and with less than 128 KB, complying with the the challenge's submission restrictions.

Additional context metrics for comparison between the base

Table 1: Experimental testing results obtained from the e2e CNN used for our DCASE2021 submission, with further pruning (LT) and quantization (INT8) optimizations.

Model	Accuracy	Params	Memory KB
DCASE2021 Baseline	46.40%	–	90.00
e2e CNN	61.49%	319,093	1,246.45
e2e CNN LT	58.73%	127,637	498.58
e2e CNN INT8	**56.50%**	**127,637**	**124.64**

Table 2: Optimization metrics used to compare the e2e CNN base models with the low-memory implementations.

Model	Memory Reduction	Acc drop	Format
e2e CNN LT	2.5X	2.76%	FP32
e2e CNN INT8	10.0X	4.99%	INT8

model and the low-memory implementation are presented in Table 2. These results present some expected insights. The resulting testing accuracy of the lottery ticket approach degrades a 2.76% accuracy from the original base model with a 2.5X memory reduction; further on, the INT8 quantized model suffers a drop of 4.99% accuracy from the original base model with a 10.0X memory reduction.

4. CONCLUSIONS

In this work, we present a low-memory implementation of an e2e CNN trained for acoustic scene classification as defined in the DCASE2021 Task 1a challenge guidelines. By exploring transfer learning, pruning, and quantization to execute neural networks model optimization, we were able to successfully construct an end-to-end audio classification deep learning-based model that achieves 56.50% accuracy performance on the DCASE2021 testing dataset, with 124.64 KB of memory size.

5. REFERENCES

[1] I. Martín-Morató, T. Heittola, A. Mesaros, and T. Virtanen, "Low-complexity acoustic scene classification for multi-device audio: analysis of dcase 2021 challenge systems," 2021.

[2] T. Heittola, A. Mesaros, and T. Virtanen, "Acoustic scene classification in dcase 2020 challenge: generalization across devices and low complexity solutions," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020)*, 2020, submitted. [Online]. Available: https://arxiv.org/abs/2005.14623

[3] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017.

[4] P. Lopez-Meyer, J. A. del Hoyo Ontiveros, H. Lu, and G. Stemmer, "Efficient end-to-end audio embeddings generation for audio classification on target applications," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 601–605.

[5] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *CoRR*, vol. abs/1710.09412, 2017. [Online]. Available: http://arxiv.org/abs/1710.09412

[6] J. Frankle and M. Carbin, "The lottery ticket hypothesis: Training pruned neural networks," *CoRR*, vol. abs/1803.03635, 2018. [Online]. Available: http://arxiv.org/abs/1803.03635

[7] A. D. Kozlov, I. A. Lazarevich, V. Shamporov, N. Lyalyushkin, and Y. Gorbachev, "Neural network compression framework for fast model inference," *ArXiv*, vol. abs/2002.08679, 2020.

[8] "Neural network compression framework for pytorch (nncf)," 2020. [Online]. Available: https://https://github.com/openvinotoolkit/nncf_pytorch