DCASE 2020 CHALLENGE TASK 1B: LOW-COMPLEXITY CNN-BASED FRAMEWORK FOR ACOUSTIC SCENE CLASSIFICATION

Technical Report

Lam Pham\textsuperscript{1*}, Dat Ngo\textsuperscript{2†}, Huy Phan\textsuperscript{3‡}, Ngoc Q. K. Duong\textsuperscript{4§},

\textsuperscript{1} University of Kent, UK
\textsuperscript{2} Ho Chi Minh University of Technology, Vietnam
\textsuperscript{3} Queen Mary University of London, UK
\textsuperscript{4} InterDigital R&D France, France

ABSTRACT

This report presents a low-complexity CNN-based deep learning framework for acoustic scene classification task (ASC). The framework uses time-frequency representation (i.e. spectrogram) referred to as front-end feature extraction. The extracted spectrograms are fed into a CNN-based architecture, referred to as the baseline, for classification. Next, quantization and pruning techniques are applied on the pre-trained baseline to finetune and further compress the network size, eventually resulting in low-complexity models with competitive performance.

Index Terms— Convolutional Neural Network (CNN), pruning, quantization, mixup data augmentation, spectrogram, Gammatone filter.

1. INTRODUCTION

Deep Learning has become a mainstream approach for various research fields such as computer vision, natural language processing, and recently emerging research field named “machine hearing” [1]. Applied to acoustic scene classification (ASC), one of main tasks of “machine listening”, CNN-based network architectures have surpassed human performance [2]. However, the state-of-the-art systems have come at an increasing cost of computation due to complex models used, making them infeasible for edge applications. Indeed, the summary of system characteristics [3] reported in the recent DCASE 20219 indicated that almost the architectures used in top ten submissions exceeded 6 M non-zero parameters. Some systems presented even much more complex models that have more than 100 M non-zero parameters. To deal with this challenge, model compression techniques have drawn increasing attention in recent years. Two main approaches of compression are quantization and pruning. Recently, TensorFlow framework 2.0 provides a complete guide for both the compression methods mentioned in [4]. Though such model compression techniques have been widely studied in machine learning and computer vision communities, they have less investigated for audio tasks.

In this report, we firstly propose a deep learning framework with low-complexity CNN-based model for the ASC task, referred to as the baseline. Next, we adopt the quantization and pruning techniques to further compress and fine-tune the pre-trained CNN baseline to obtain a small-footprint model. We use the DCASE 2020 Task 1B dataset to evaluate the framework with/without using these compression techniques and compare their performance to it of the DCASE baseline.

2. DCASE 2020 TASK 1B DATASET

The DCASE 2020 Task 1B dataset [5] was recorded by a single device namely A with binaural channel and sample rate of 48kHz. The dataset comprises of 10 acoustic scenes that are grouped into three main contexts: indoor (airport, metro-station, and shopping-mall), outdoor (park, public-square, street-pedestrian, street-traffic), and transportation (bus, metro, tram). In this report, we obey DCASE 2020 challenge to separate development set into training and test subsets used for training and testing processes, respectively. The accuracy on the test subset is then reported.

3. CNN-BASED FRAMEWORK ARCHITECTURE

![Figure 1: The high-level architecture and processing sequence of the proposed framework.](image)

The proposed framework is described in Fig. 1. Initially, raw audio signal from the channel 1 is transformed into Gammatone spectrogram (Gamma) [6] with parameters summarized in Table 1. Then, mixup data augmentation [7, 8] is applied on entire spectrograms of $128 \times 428$ to generate new spectrograms. Next, the mixup spectrograms are fed into the CNN-based network.

The CNN-based network configured as Table 2 comprises four Conv. blocks and one Dense block, which are performed by Convolutional layer (Cv[kernel size]), Rectified Linear Unit (Relu), Batch

\footnotesize
\*ldp7@kent.ac.uk
\†thanhdat5494@gmail.com
\‡h.phan@qmul.ac.uk
\§quang-khanh-ngoc.duong@interdigital.com
normalization (Bn), Max pooling (Mp[kernel size]), Global max pooling (Gmp), Drop out (Dr(Drop ratio)), Fully connected layer (Fl), and Softmax layers.

After training the CNN-based network, two compression techniques, 8-bits training-aware quantization and pruning mentioned in Tensorflow Model Optimization Toolkit [4], are applied to fine-tune the pre-trained CNN-based network. The resulting model has lower complexity, but remains competitive performance.

4. HYPERPARAMETER SETTING

The CNN-based network was implemented using Keras framework. Network training made use of the Adam optimizer [9] with 100 training epochs, a mini batch size of 100. The KL-divergence loss reads

\[ L_{KL}(\theta) = \frac{1}{N} \sum_{n=1}^{N} y_n \log \left( \frac{y_n}{\hat{y}_n} \right) + \frac{\lambda}{2} ||\theta||^2. \]

where \( \theta \) denotes the trainable network parameters and \( \lambda \) denotes the \( \ell_2 \)-norm regularization coefficient, set to 0.0001. \( N \) is the number of training samples, \( y_i \) and \( \hat{y}_i \) denote expected and predicted results, respectively.

5. EXPERIMENTAL RESULTS

The obtained results are shown in Table 3. Without compression, the proposed CNN outperforms the DCASE baseline, improving the accuracy by 5.7% absolute. Compressing the CNN network using 8-bits quantization and pruning techniques, we achieve the compressed networks with a model size 4 and 2 times smaller than the original model, 61.5 KB and 122.8 KB, respectively, at the cost of decreasing accuracy. Compared to the DCASE baseline, the compressed models have 7.4 and 3.7 times smaller footprints but still achieve better performance, 91.9% and 90.5%, respectively, compared to 87.3% obtained by the baseline.

6. CONCLUSION

In this work, we have investigated a CNN-based framework with a small number of parameters for the DCASE ASC task. Thank to quantization and pruning techniques supported by Tensorflow framework, the proposed network is further compressed to have less non-zero parameters, but still outperforms the DCASE 2020 Task 1B baseline. Future work would be devoted to investigate network distillation techniques [11] where the original CNN model could be used to guide the fine-tuning of the compressed and quantized models.

7. REFERENCES


Table 1: Setting of spectrogram transformation.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spectrogram</td>
<td>Gammatone</td>
</tr>
<tr>
<td>Window size</td>
<td>2048</td>
</tr>
<tr>
<td>Hop size</td>
<td>1024</td>
</tr>
<tr>
<td>The number of FFT points</td>
<td>4096</td>
</tr>
<tr>
<td>The number of filter banks</td>
<td>128</td>
</tr>
<tr>
<td>Min frequency</td>
<td>10 Hz</td>
</tr>
</tbody>
</table>

Table 2: CNN-based network architecture

<table>
<thead>
<tr>
<th>Architecture</th>
<th>layers</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conv. Block 01</td>
<td>Bn - Cv [3×3] - Relu - Bn - Mp [2×2] - Dr (20%)</td>
<td>128x128</td>
</tr>
<tr>
<td>Conv. Block 02</td>
<td>Bn - Cv [3×3] - Relu - Bn - Mp [2×2] - Dr (20%)</td>
<td>64x107x32</td>
</tr>
<tr>
<td>Conv. Block 03</td>
<td>Bn - Cv [1×1] - Relu - Bn - Mp [2×4] - Dr (30%)</td>
<td>32x54x64</td>
</tr>
<tr>
<td>Conv. Block 04</td>
<td>Bn - Cv [1×1] - Relu - Bn - Gmp - Dr (35%)</td>
<td>16x13x128</td>
</tr>
<tr>
<td>Dense Block</td>
<td>Fl - Softmax layer</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 3: Performance compared to DCASE 2020 Task 1B baseline

<table>
<thead>
<tr>
<th>System</th>
<th>Acc.(%)</th>
<th>Non-zero para. (KB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCASE 2020</td>
<td>87.3</td>
<td>430.0</td>
</tr>
<tr>
<td>CNN network</td>
<td>93.0</td>
<td>245.5</td>
</tr>
<tr>
<td>CNN network w/ quantization</td>
<td>91.9</td>
<td>61.5</td>
</tr>
<tr>
<td>CNN network w/ pruning</td>
<td>90.5</td>
<td>122.8</td>
</tr>
</tbody>
</table>

In this work, we have investigated a CNN-based framework with a small number of parameters for the DCASE ASC task. Thank to quantization and pruning techniques supported by Tensorflow framework, the proposed network is further compressed to have less non-zero parameters, but still outperforms the DCASE 2020 Task 1B baseline. Future work would be devoted to investigate network distillation techniques [11] where the original CNN model could be used to guide the fine-tuning of the compressed and quantized models.