EIW 2022 - Challenge

May 2022

Abstract

This document presents our approach to the challenge presented at the Edge Intelligence Workshop 2022, in which the goal is to train a neural network during 10 minutes on a small datasets consisting of 5000 training images and test its accuracy on a test set of 1000 images. We used a scaled SEnet model combined to a data augmentation process and a hyperparamater optimization. Our proposed network achieves an average accuracy of around 78% on a subset of CIFAR-10 and ten different subsets of Mini-Imagenet. The link to our github (for the code) : `https://github.com/LouisDage/Edge-Intelligence-Workshop-Challence.git`

1 Introduction

Neural networks are becoming increasingly big since a lot of progress are made each year in order to improve their performances. However, this creates a substantial requirement for computing capacity and many devices are not able to bear this demand even for the inferences. Therefore, reducing the size of networks is an important goal. Edge Intelligenche Workshop Challenge addresses this issue and consists in training a network on data similar to CIFAR-10 under the following constraints: the training lasts exactly 10 minutes and it is done on a Nvidia V100 GPU with 32GB memory, running with an Intel(R) Xeon(R) Gold 6230 CPU @ 2.10GHz processor, with 12GB of RAM. The training is made on a 10 classes dataset with only 5000 32x32 RGB images. Considering this setting, we identify two challenges. The first one is the time of training which is considerably smaller than usual. The second one is the small dataset because models tend to overfit with a such little dataset. To resolve these problems, we opted for a network with few parameters to reduce the computational cost and to use data augmentation to increase the dataset size.

2 Proposed approach

2.1 Baseline Model

We used a SEnet model proposed in [1]. The idea of a SEnet model is to use "Squeeze-and-Excitation" blocks in order to recalibrates channel-wise feature responses by explicitly modelling interdependencies between channels. The network we used has only 6356039 parameters.

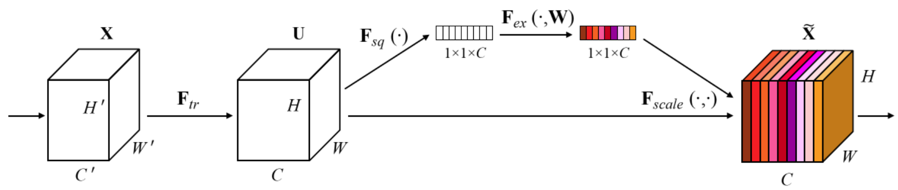

Figure 1: SEnet Block Architecture. Image taken from [1].

2.2 Data Augmentation Method

Given the small size of the dataset, we observed that our candidate model is prone to overfitting. Since the dataset is tiny, the model does not gain enough performance during training by using the basic techniques of image augmentation. We used the AutoAugment [5] approach, which is an automatic way of selecting optimal data augmentation policies. Also, we added the CUTOUT [6] technique which consists of masking out random sections of input images during training. Finally we used the CIFAR-10 Policy method that randomly choose among 25 Sub-policies one that modify the image (which was a lot more efficient than Imagenet Sub-policies) .

2.3 NAS & Hyper Parameter Optimization

We scale the depth and width of the SENet blocks non-uniformly. Each one of the four "PreActBlock" is multiplied by a depth coefficient $d_i \in [0.4, 3]$ and a width coefficient $w_i \in [0.1, 2]$. The NAS is carried out by the NOMAD Algorithm [7] Our results are reported in Table 1.

Table 1: Results of the Architecture Research

Parameter	d_1	d_2	d_3	d_4	w_1	w_2	w_3	w_4
Value	2.05	1.97	1.88	3	0.78	1.22	0.80	0.67

After we've found the architecture we ran the NOMAD Algorithm [7] in order to perform a Hyper-Parameter Optimization (HPO) and find an optimal training regime. We set our search space as four optimizers (SGD, Adam, Adagrad, RMSProp) and for each optimizer we consider four hyperparameters as listed in Figure 2. The best solution found by the algorithm is reported in Table 2.

Hyperparameters related to the training of the VAE.

Optimizer	Hyperparameter	Type	Range
Stochastic Gradient Descent (SGD)	Initial learning rate	Real	[0;1]
	Momentum	Real	[0;1]
	Damping	Real	[0;1]
	Weight decay	Real	[0;1]
Adam	Initial learning rate	Real	[0;1]
	β_1	Real	[0;1]
	β_2	Real	[0;1]
	Weight decay	Real	[0;1]
Adagrad	Initial learning rate	Real	[0;1]
	Learning rate decay	Real	[0;1]
	Initial accumulator	Real	[0;1]
	Weight decay	Real	[0;1]
RMSProp	Initial learning rate	Real	[0;1]
	Momentum	Real	[0;1]
	Smoothing constant	Real	[0;1]
	Weight decay	Real	[0;1]

Figure 2: HPO search space. Image taken from [4].

Table 2: Results of the HPO

Optimizer	learning rate	momentum	weight decay	dampening
SGD	0.042	0.9	0.005	0

3 Experiment and Results

Once we found the best solution both for the architecture and the hyperparameters we tested on a random shuffled subset of CIFAR-10 composed by 5K training images and 1k images of validation and on ten different subset of Mini-Imagenet with the same composition and we averaged the results. We trained on a Nvidia A100 GPU with 40GB memory during 600 seconds. Our results are reported on Table 3.

	Accuracy(%)	
Time	Train	Valid
600s	83.18	78.1

Table 3: Results of experiments

4 Conclusion

The Edge Intelligence workshop Challenge focuses on training a neural network under low resources, more precisely limited data and training time. In order to train a model efficiently, we proposed an approach that finds a classifier that performs accurately on a tiny dataset and a limited training time budget. We used known blocks, SEnet blocks, in order to create a new network with the NOMAD software, which implements a derivative-free optimization algorithm for the neural Architecture search. Then we re-used NOMAD, for the HPO. Also, we used data augmentation techniques to improve the performance of the SEnet model. Finally we have reached 78.1% accuracy on average on a subset of CIFAR-10 and ten hand-made subset of Mini-Imagenet.

References

[1] Jie hu, Li Shen, Samuel Albanie, Gang Sun and Enhua Wu. Squeeze-and-Excitation Networks https://arxiv.org/pdf/1709.01507v4.pdf

[2] C. Audet and J.E. Dennis, Jr. Mesh Adaptive Direct Search Algorithms for Constrained Optimization.SIAMJournalonOptimization, 17(1):188–217, 2006. doi: 10.1137/040603371.

[3] Dounia Lakhmiri, Mahdi Zolnouri, Christophe Tribes, Sébastien Le Digabel Eyyüb Sari. Efficient Training Under Limited Ressources.

[4] Dounia Lakhmiri, Ryan Alimo, Sébastien Le Digabel, Anomaly detection for data accountability of Mars telemetry data

[5] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment:Learning augmentation policies from data.arXiv preprint arXiv:1805.09501, 2018

[6] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. arXiv preprint arXiv:1708.04552, 2017.

[7] S. Le Digabel. Algorithm 909: NOMAD: Nonlinear Optimization with the MADS algorithm.ACM Transactions on Mathematical Software, 37(4):44:1–44:15, 2011. http://dx.doi.org/10.1145/1916461.1916468