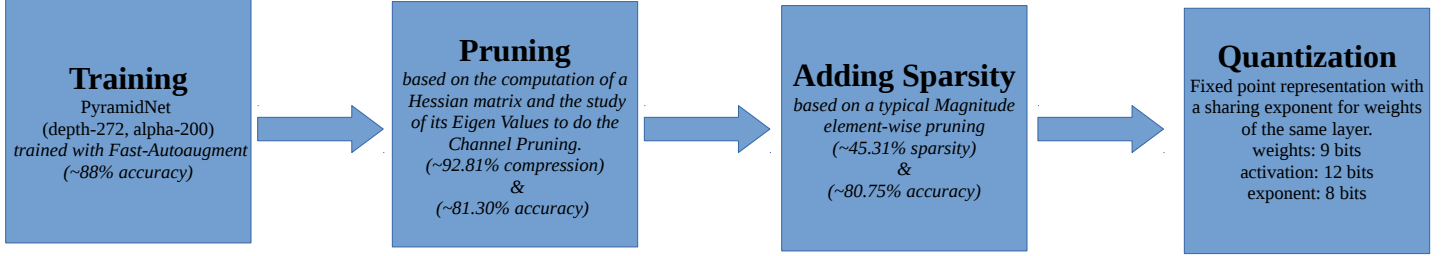


Micronet Challenge Report:

first submission & second submission

Here is a schema describing our first submission:



This **first** submission gives us a final micronet score of : **0.0467**. For a corresponding accuracy of **80.15%**.
This **second** submission gives us a final micronet score of : **0.0456**. For a corresponding accuracy of 80.47%.

The only difference between the two submission is the fact that in the second one we merge the BatchNorm layers with the Convolutional layers.

To compute the micronet score we followed the following steps.

Storage:

$$\text{Storage} = \frac{\text{TotalNonZeroParameters} * \text{ParamBits}}{32} + \frac{\text{TotalParameters}}{32}$$

Math Ops:

$$\text{MathOps} = \frac{\text{TotalFlopMults} * \text{MultBits}}{32} + \frac{\text{TotalFlopAdds} * \text{AddBits}}{32} + \frac{\text{TotalFlopExponent} * \text{ExponentBits}}{32}$$

With:

- $\text{ParamBits} = 9$
- $\text{MultBits} = 12$
- $\text{AddBits} = 32$
- $\text{ExponentBits} = 8$

The values TotalFlopMults and TotalFlopAdds were computed following the code given by the micronet challenge. Please see the file *Micronet/Test/compute_flops.py* for implementation details.

Score:

$$\text{score} = \frac{\text{MathOps}}{10.49 B} + \frac{\text{Storage}}{36.5 M}$$

Please see the file *Micronet/Test/main.py* for overall implementation details

If one wants to reproduce all the results, please follow the indications given in the different README files.
The order of execution should be :

- 1/ Training
- 2/ Pruning
- 3/ Sparsity
- 4/ Quantization

5/ Test

To only obtain the final micronet score run the file *MicroNet/Test/main.py* without changing anything.