Machine Learning for IoT - Homework 02

Francesco Di Salvo s282418

Francesco Lacriola s292129

Gianluca La Malfa s290187

Exercise 1 - Multi-Step Temperature and Humidity Forecasting

The goal of the exercise was to train multi-output models for temperature and humidity forecasting to infer multi-step predictions. We combined three different optimization techniques: *structured pruning* via width scaling, *magnitude base pruning* with polynomial decay and *post training quantization*. Finally, in order to reduce the model size, we compressed them through *zlib*. We had to satisfy two set of constraints:

- with an output step equals to 3, the best configuration used a Multi Layer Perceptron trained for 22 epochs with adam optimizer. Moreover, we used a width scaler (α) equal to 0.07 in order to reduce the number of units per dense layers, shifting from 128 to 9. After that, a magnitude base pruning optimization has been applied with initial and final sparisity parameters equal to 0.30 and 0.85. Once we have such a sparse model, any compression algorithm can reduce its disk occupation. Finally, we employed a weights only PTQ optimization in order to memorize the values with an int8 representation instead of the default float32. With that being said, we obtained a compressed model size equal to 1.44 kB, 0.299 and 1.16 as MAE for temperature and humidity, respectively.
- with an output step equals to 9, it will be harder to make good precisions as before, therefore we trained a Multi Layer Perceptron for 30 epochs, with $\alpha = 0.08$, and 0.3 and 0.85 as sparisity coefficients. Finally, a weights only PTQ optimization, as before. So, we obtained 0.68 and 2.36 as temperature and humidity MAE and a final compressed size of 1.69 kB.

Exercise 2 - Keyword Spotting

The goal of the exercise was to train models for keyword spotting on the original mini speech command dataset. In this work we combined optimization techniques both for the model and the audio files. In fact, in order to reach the constraints in B and C we performed a resampling phase to decrease the total latency of the preprocessing step and as a consequence also the size of the model. For all the versions we used MFCC approximation because of the need of a high accuracy of the model. We used structured pruning and weights only post PTQ. In order to satisfy all the constraints we employed the following settings:

- For version *a*, we used a DS-CNN, trained for 25 epochs, using an Adam optimizer with a learning rate equals to 0.005, α equals to 0.85 and a weights only PTQ. Moreover, the MFCC frame length and step were 640 and 320, respectively. We obtained a compressed model size of 88 kB and a satisfactory accuracy of 93.75%.
- For both versions *b* and *c*, we used a DS-CNN, trained for 25 epochs, using an Adam optimizer with a learning rate equals to 0.005, α equals to 0.3 and a weights only PTQ. Moreover, in order to reduce the inference latency we resampled the audio to 8000 Hz and the MFCC frame length and step that performed the best were 240 and 120, respectively. We obtained a compressed model size of 22.54 kB, an accuracy of 91.75% and a total inference latency of 32.81 ms.