Machine Learning for IoT - Politecnico di Torino

Homework 3 report

Paolo Aberto
StudentID : s278098

Lorenzo De Nisi
Student ID: s276545

Carmine De Stefano
Student ID: s278176

1 Big/Little Inference

For the big/little inference, we chose to use perform it with the use of REST, since it was the best choice when working with a single client and a server. When training the big model we used MFCC to maximize the accuracy, while we switched to STFT in the little one, in order to reduce significantly the inference time.
For both networks, we decided to use a DS-CNN model, increasing the number of convolutional filters in the big model. For the small model, we also used magnitude based pruning, in order to respect thesize constraints while mantaining a good accuracy.
We also implemented learning rate policies for both the networks during training.

We tried different success checker policies, like a treshold on the entropy of the probabilities array. We finally settled on the **score margin** policy, where we used a threshold of 0.58 on the difference between first and second biggest scores (probabilities) coming from the last network layer.

When it comes to the communication cost, we managed to call the server for a prediction 113 times with a total cost of about 4.47 MB and a final accuracy of 93.125 %. It is worth saying that we forced the little model to use it's own predictions when we were close to the limit (4.5 MB), even if the confidence was below the threshold.

Model	Size	Compressed size	Epochs	MFCC accuracy	Test set
Big	2,2 MB	2,1 MB	35	YES	95.125%
Little	28,5 kB	19,2 kB	35	NO	90.620%

Table 1: Models used for the big/little inference

2 Cooperative Inference

For the cooperative inference, the chosen number of models is 4.
The communication is performed through MQTT, considering that it is relatively easy to publish the recording once for all the, possibly many, models with respect to create a web service for each of them. Two topics are used, one for the recordings (`276545/recording`) and one for the inferences (`276545/predictions`).
We managed to keep the number of models as low as possible while preserving accuracy. As expected all the models have an individual accuracy that is lower than the cooperative final accuracy that is 95.13% .
Two of them are derived from the proposed DS-CNN, while the remaining two are derived from the proposed CNN, with some modification on the BatchNorm layer and filters and biases of the Conv2D.
The implementation relies on queues both on each device and on the cooperative client, used to store received messages. In that way we do not wait the answers for each recording before sending the next one and this results in shorter execution time.
A timeout policy is implemented to handle missing messages that can be lost, and the QOS is set to 0.
The cooperative policy consists of averaging the logits (output of the last layer of the models) and taking the argmax of them.

Ver.	Model	Modification w.r.t. proposed models	Epochs	lr	Test set accuracy
1	CNN-0	-	20	0.01	94.250
2	CNN-1	Conv2D(filters=64, bias=True), BatchNormalization(momentum=0.2)	20	0.01	93.125
3	DS-CNN-0	-	20	0.01	93.625
4	DS-CNN-1	Conv2D(filters=128, bias=True), BatchNormalization(momentum=0.2)	20	0.01	92.500
	Cooperative inference				**95.13**

Table 2: Models used for the cooperative inference