

1 Problem description

In this home assignment, we will implement the model optimized during the home assignment 2 on a STM32F4 board.

2 Description of the model

The model I have converted to a `.tflite` is a bit different from the model that I used in the home assignment 2, as the activation function *gelu* was not supported by the converted, so I used *relu* instead.

The model has 3 hidden layers of, respectively, 110, 100, 90 neurons. The learning rate is 0.05, it is trained on 30 *epochs* and the optimizer is *adam*.

3 Converting the model to .tflite

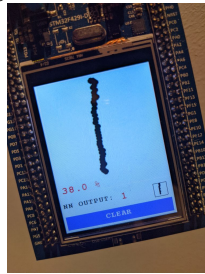
Since I had the script to generate the model, I converted directly the model using `TFliteConvert.from_keras_model`. I had to enable `SELECT_TF_OPS` for it to work, and also to change my activation function because it made the converter crash.

4 Running the model on the board

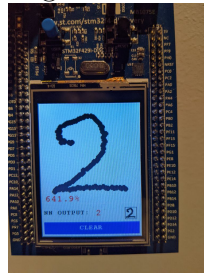
Using the lab tutorial, I have been able to download my model on the board. Luckily, my model was adapted to the board already.

After some bug solving, I was able to load my model from the assignment 2 on the board, and to predict some numbers with it.

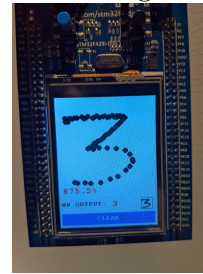
Digit classification using model from home assignment 2



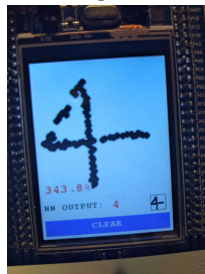
Digit 1



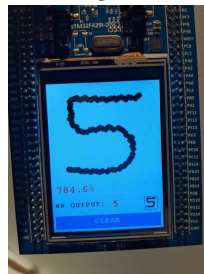
Digit 2



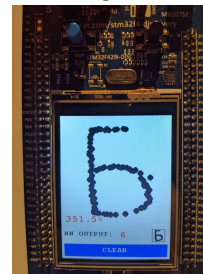
Digit 3



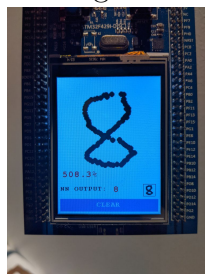
Digit 4



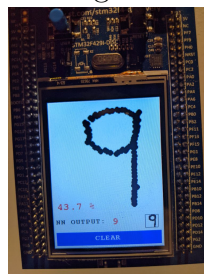
Digit 5



Digit 6



Digit 8



Digit 9

The model was able to recognize most of the digit, but it may have been over-fitted a bit since some variation of number are not well recognized: 1 drawn with the little line at the top, 9 with a line on the bottom, or 4 without closing the vertical line, for exemple.

The digit 7 was the most problematic one; and it is a challenge to find a way where it is well classified.

5 Issues

5.1 Running the model

As said before, my model is probably a bit over-fitted. It was hard to detect it before, as to test it on my computer I had to draw my digit each time, then save the image, then load it. It was much more easier to test a lot of data on the board, and therefore to see what would prevent a digit to be classified correctly.

I have a problem with the display of the accuracy : half of the time, the accuracy goes wild and is printed as more than 100%, which is obviously impossible. I have searched, but not found the reason for that. The weirdest thing is that sometimes, the accuracy seems to be displayed correctly (digit 1 or 9 on the pictures).

5.2 Putting the model on the board

I had a lot of issues trying to put the model on the board.

Some of them were referenced in the *lab issues* page on Moodle, which saved me a lot of time : *multiple undefined references to ai_platform_**, *error finding ai_platform.h*, *problem with code generation in cubeMX*, and *Board LCD is shaking and not initialized correctly*.

I also had a lot of struggle with the version of *X-CUBE-AI*: each time I tried to generate a new code I had to uninstall and reinstall it, as it keep selecting the 7.0 version and I could not use 5.0 version if the 7.0 was also installed.