097200 - Deep learning: Multimodal problems

Submission date: 27/12/2020

1. Exercise

In the following exercise, you will deal with a multimodal problem, specifically, visual question answering. You should write your own training code and meet the following constraints.

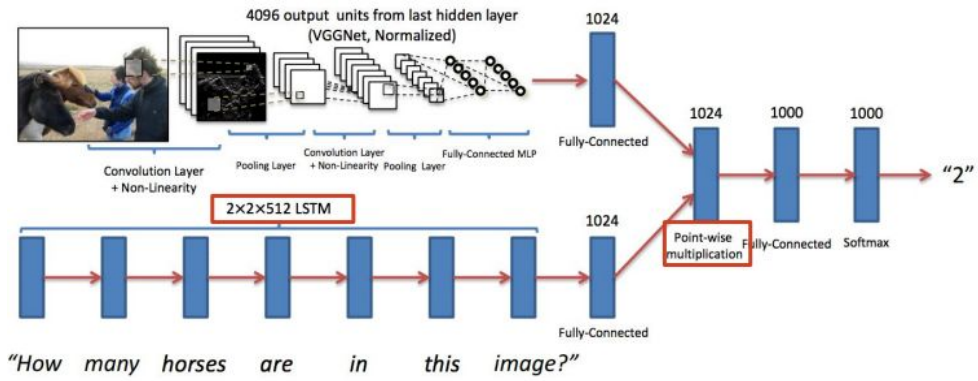
2. Classification Network

Write a complete training procedure for a visual question answering on VQA 2.0 (https://visualqa.org/). The data is mounted to your machine under /datashare

Design and train your network so that it will satisfy the 2 following goals:

1. Final accuracy on the val-set should be **> 45%**
2. You can use as many parameters as you want

A simple baseline network:



High-level guidelines:

1. Write a custom Dataset (Tutorial 3) for VQA (*getitem* function should return a tuple of image, question, and answer)
2. Build a multi-modal model that gets§ image and question as input and produces probabilities for each answer:
    a. As a simple baseline, you can concatenate each modality's hidden representations and then forward it through a fully-connected network.
    b. To improve it, you may use transformers, attention, and any other creative ideas.
3. Train and evaluate your model using the accuracy measure

You can use the following template: https://github.com/itaigat/pytorch-template

3. Submission instructions

Submission will be in pairs (course partners) and will contain a short (two pages) pdf report containing:

1. Model architecture description and illustration, training procedure (hyper parameters, optimization details etc.).
2. Two convergence plots - (1) for error (2) for loss, both as a function of time (epochs). Each plot should depict both training and val performance (i.e. two curves per plot, one for train and one for val).
3. Summary of your attempts and conclusions. Your conclusions and explanations should be based on the actual results you received during your attempts.
4. Your best val accuracy result (of the submitted model) should be written explicitly in your report.

In addition, you should supply:

1. Code (python file) able to reproduce your results - we might test it on different variants on these datasets.
2. The trained network with trained weights (.pkl file). If the model size is less than 500MB you should submit it on the Moodle. Otherwise, upload it to your Google-Drive.
3. A function called "*evaluate_hw2()*". The function should load the VQA 2.0 validation set, load your trained network (you can assume that the model file is located in the script folder) and return the average accuracy on the val-set. This function should be written in a separate script. Use this line to load your model:
4. model.load_state_dict(torch.load('model.pkl',map_location=lambda storage, loc: storage))

<u>Moodle submission:</u>

you should submit a Zip (not rar!) file containing:
- **Code – as many files you need (one of them should be "main.py" and will contain the running process)**
- **1 pdf file (containing your names and ID's)**
- **.pkl file (If the file is too big for the Moodle, upload it to your Google-Drive and copy the link to your pdf report)**

4. Grades policy

1. Successful submission – 40 points.
2. Report - 40 points.
3. Competition - 20 points: the teams will be sorted according to val error results, then will be split into groups. The group with the best performance will receive 20 – 0*(20/#groups), the next best group will receive 20 – 1*(20/#groups) and so on. Alternatively, the formula will be 20*((accuracy - min)/(max - min)). The formula that gets the best result for you (on average) will be chosen.

<u>Good luck!</u>