

## HOMework 3

16824 VISUAL LEARNING AND RECOGNITION (SPRING 2023)

<https://piazza.com/class/lcy4ow5l5xp2f1>

RELEASED: Sun, 19th March 2023

DUE: Mon, 3rd April 2023

Instructor: Deepak Pathak

TAs: Ananye Agarwal, Rohan Choudhury, Murtaza Dalal, Russell Mendonca

### START HERE: Instructions

- **Collaboration policy:** All are encouraged to work together BUT you must do your own work (code and write up). If you work with someone, please include their name in your write-up and cite any code that has been discussed. If we find highly identical write-ups or code or lack of proper accreditation of collaborators, we will take action according to strict university policies. See the [Academic Integrity Section](#) detailed in the initial lecture for more information.
- **Late Submission Policy:** There are a **total of 7** late days across all homework submissions. Submissions more than 7 days after the deadline will receive a 0.
- **Submitting your work:**
  - We will be using Gradescope (<https://gradescope.com/>) to submit the Problem Sets. Please use the provided template only. Submissions must be written in LaTeX. All submissions not adhering to the template will not be graded and receive a zero.
  - **Deliverables:** Please submit all the `.py` files. Add all relevant plots and text answers in the boxes provided in this file. TO include plots you can simply modify the already provided latex code. Submit the compiled `.pdf` report as well.

*NOTE: Partial points will be given for implementing parts of the homework even if you don't get the mentioned numbers as long as you include partial results in this pdf.*

# 1 Image Captioning with Transformers (70 points)

We will be implementing the different pieces of a Transformer decoder ([Transformers](#)), and train it for image captioning on a subset of the [COCO dataset](#).

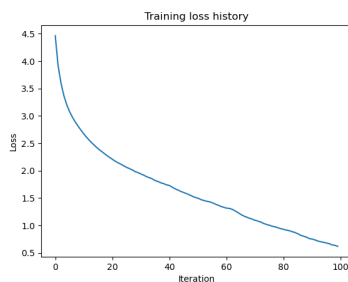
- **Setup:** Run the following command to extract COCO data, in the `transformer_captioning/datasets` folder: `./get_coco_captioning.sh`
- **Question:** Follow the instructions in the `README.md` file in the `transformer_captioning` folder to complete the implementation of the transformer decoder.
- **Deliverables:** After implementing all parts, use `run.py` for training the full model. The code will log plots to `plots`. Extract plots and paste them into the appropriate section below.
- **Expected results:** These are expected training losses after 100 epochs. Do not change the seed in `run.py`.
  - 2-heads, 2-layers, lr 1e-4: Final loss  $\leq 1$
  - 4-heads, 6-layers, lr 1e-4: Final loss  $\leq 0.3$
  - 4-heads, 6-layers, lr 1e-3: Final loss  $\leq 0.05$

1. Paste training loss plots for each of the three hyper-param configs

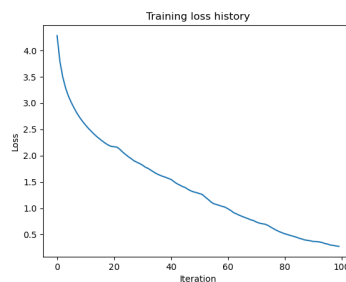
2-heads-2-layers-lr-1e-4: **0.619738**

4-heads-6-layers-lr-1e-4: **0.273871**

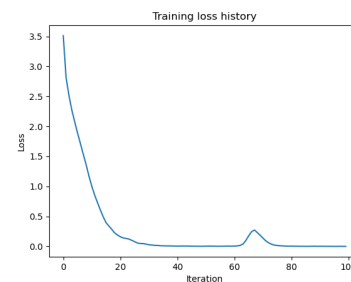
4-heads-6-layers-lr-1e-3: **0.001201**



(a) 2-heads-2-layers-lr-1e-4



(b) 4-heads-6-layers-lr-1e-4



(c) 4-heads-6-layers-lr-1e-3

2. Paste any three generated captioning samples from the training set. The provided code creates these plots at the end of training.

train  
a train at the station and people waiting <END>  
GT:<START> a train at the station and people waiting <END>



(a) Sample1

train  
a <UNK> blurry image a decorated fridge <END>  
GT:<START> a <UNK> blurry image a decorated fridge <END>



(b) Sample2

train  
two large <UNK> <UNK> <UNK> in a dirt pasture <END>  
GT:<START> two large <UNK> <UNK> <UNK> in a dirt pasture <END>

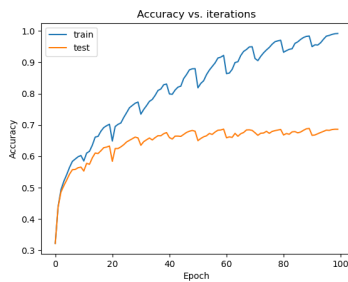


(c) Sample3

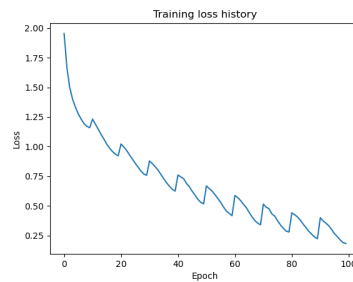
## 2 Classification with Vision Transformers (30 points)

We will use the transformer you implemented in the previous part to implement a Vision Transformer (ViT), for classification on CIFAR10.

- **Question:** Follow the instructions in the `README.md` file in the `vit_classification` folder. You are encouraged to reuse code from the previous question.
- **Deliverables:** Run training using `run.py` for training the full model. The code will log plots `acc_out.png` (train and test accuracy) and `loss_out.png` (train loss).
- **Expected Results:** After 100 epochs, test accuracy should be  $\approx 68\%$ , train accuracy should be  $\approx 100\%$ , and training loss  $\approx 0.25$ .



(a) Train/test accuracy



(b) Training loss

```
Train accuracy: 0.985680
(epoch 97 / 100) loss: 0.213466
Test accuracy: 0.685500
Train accuracy: 0.988960
(epoch 98 / 100) loss: 0.191902
Test accuracy: 0.686500
Train accuracy: 0.991040
(epoch 99 / 100) loss: 0.184173
Test accuracy: 0.686400
Train accuracy: 0.991780

Process finished with exit code 0
```

(c) Losses printed

**Collaboration Survey** Please answer the following:

1. Did you receive any help whatsoever from anyone in solving this assignment?

☒ Yes

☐ No

- If you answered 'Yes', give full details:
- (e.g. "Jane Doe explained to me what is asked in Question 3.4")

I discussed the concept of transformers with Ishwaryaa S.V.

2. Did you give any help whatsoever to anyone in solving this assignment?

☒ Yes

☐ No

- If you answered 'Yes', give full details:
- (e.g. "I pointed Joe Smith to section 2.3 since he didn't know how to proceed with Question 2")

I discussed the concept of transformers with Ishwaryaa S.V.

3. Note that copying code or writeup even from a collaborator or anywhere on the internet violates the [Academic Integrity Code of Conduct](#).