IITM-CS7016 : Topics in Deep Learning — Given on: Sept 27, 11:55pm

Programming Assignment (Individual) — Due on : October 23, 11:55pm IST

- The aim of this assignment is to train and test Faster RCNN, RetinaNet and YOLO-v3 object detection model on DVQA dataset.
- Collaborations and discussions with others are strictly prohibited.
- This assignment is going to be time-consuming. **PLEASE START EARLY**.
- It will be better if you use **Caffe2/Tensorflow/Pytorch** library (Python) for your implementation. If you are using any other languages, please contact the TAs before you proceed.
- Note that you can use the publicly available code to implement the models. Citing the public APIs is **COMPULSORY**.
- You have to turn in the well documented code along with a report with **Detailed Observations** and inferences of the results electronically in Moodle.
- Typeset your report in Latex code provided (attached). It is necessary to fill '**Checklist**' in the attached sample report. Reports which are not written using Latex will not be accepted.
- The report should be precise and concise. Unnecessary verbosity, (like **Theory about object detection models**) will be penalized.
- You **MUST** run *sanity_check.py* script (attached) on your submission zip file.
- You can find the evaluation pattern (marks distribution) at the last page.
- *You have to check the Moodle discussion forum regularly for updates regarding the assignment.*

1 Task

1.1 Problem Definition

In this assignment, you will train and test Faster RCNN, RetinaNet and YOLO-v3 on the mini-DVQA dataset. Specifically, the task is as follows: Given an input image from the dataset, you have to train the network to detect different objects present in the image and classify them into 1 of 9 classes *viz., bar, legend-heading, legend-label, title, xlabel, ylabel, xticklabel, yticklabel and background.*

1.2 Instructions

- Download the mini-DVQA dataset from the link: https://drive.google.com/drive/folders/1Hyf3kj0jmYUs2yYSESBBiDQVkr21r4ff?usp=sharing. The dataset contains the train split, public test split (ground truth is present) and private test split (ground truth is hidden).
- Each split contains:
  1. *png* directory which contains the images in RGBA format.
  2. *annotations* file which contains the corresponding ground truth annotations for each image in MS COCO format. The COCO format for object detection is given in the link {http://cocodataset.org/#format-data}
- Train a Faster RCNN and RetinaNet model. (You can refer/re-use the model given at https://github.com/facebookresearch/Detectron). For using the above model, you might have to use the script given in the link (https://drive.google.com/file/d/1uk8qqzcvLV7fWQ6EcavVBb0BYXxPGf5U/view?usp=sharing) to install Caffe2 on a AWS GPU instance.
- Train a YOLO-v3 model. (You can refer/re-use the model given at https://github.com/AlexeyAB/darknet).
- Train the network using the entire training dataset.
- You have to run all the 3 object detection models(Faster RCNN, RetinaNet, YOLO-v3) on the private test split and submit the corresponding .txt files for each image. The name of the text file should match the name of the image file. Each line of the text file should have the following format:

  predicted-class class-confidence x1 y1 x2 y2

  where x1, y1 are the top-left co-ordinates and x2, y2 are the bottom-right co-ordinates of the bounding box as predicted by the model, predicted-class is the class-name of the object present in that bounding box and class-confidence is the confidence score of the detected object.
- The primary evaluation metric for the task is Mean Average Precision (mAP). We will evaluate your submitted predictions and rank the assignments based on the mAP scores.

1.3 Report

Prepare a report containing the observations and inferences for all the **3 models** with the following points:

- Implement/Run the models and report their best performing parameters. [**20 + 20 + 20 marks**]

- Calculate the mAP score at IOU of 0.5 on both the public and private test split and report a table with Average Precision for every class. [**5 + 10 marks**]
- Report the mAP scores at IOUs 0.75 and 0.9 on the public test set. [**10 marks**]
- Compare the 3 models by plotting a graph with IOU threshold on the X-axis and mAP scores(in %) on the Y-axis. What can you infer from the plot? [**5 marks**]
- Replace the Focal Loss in RetinaNet with vanilla Cross Entropy loss and report the observations. [**5 marks**]
- From the above experiments, which detector do you think works best for the DVQA dataset and why? [**5 marks**]

1.4 Submission Instructions:

You need to submit the source code for the assignment. All other supporting files used for generating plots, calculating mAP, etc. should also be placed in the zip file. You need a single folder (RollNoTeamMemberA_RollNoTeamMemberB_PA2, *e.g.* CS15D201_CS14B042_PA2) containing the following:

- `run.sh (containing the best hyperparameters setting)`
- `any other python scripts that you have written`
- `'report.pdf' (in Latex) of the results of your experiments`
- `'private_preds' directory which should contain text files for the predicted class and predicted bounding box co-ordinates (one per line) for each image of the private test split.`

The zip should be named as RollNoTeamMemberA_RollNoTeamMemberB_PA2.zip, (*e.g.* CS15D201_CS14B042_PA2.zip). Note that zip file and folder name **SHOULD BE SAME**.

Please run sanity_check.py by passing your zip file as command line argument (*e.g.* python sanity_check.py CS15D201_CS14B042_PA2.zip). Only, if this doesn't throws any errors, submit your assignment.