CIS 620, Advanced Topics in Deep Learning, Spring 2024

Homework 3
Due: Monday, February 12, 11:59pm
Submit to Gradescope

Learning Objectives

After completing this assignment, you will:

- Perform object detection using a multimodal architecture
- Learn about comparison metrics of similarity in latent space

Deliverables

This is a **pair** assignment for both the written and coding portions. - **One submission per pair**

1. **A PDF with your name in the agreement**

   Copy and edit this google doc to enter your names in the Student Agreement and answer the questions mentioned below. **Don’t forget to add your team member on gradescope submission!**

2. **hw3.ipynb file with the functional code**

   Complete the coding assignment in the Jupyter Notebook and upload the file to Gradescope. **Don’t forget to add your team member on the gradescope submission!**

Note that there is a separate assignment for the papers, which will be handed in separately.

Homework Submission Instructions

Written Homeworks

All written homework must be submitted as a PDF to Gradescope. **Handwritten assignments (scanned or otherwise) will not be accepted.**

Coding Homeworks

All coding assignments will be done in Jupyter Notebooks. We will provide a .ipynb template for each assignment as well as function stubs for you to implement. You are free to use your own installation of Jupyter for the assignments, or Google Colab, which provides a Jupyter Environment connected to Google Drive along with a hosted runtime containing a CPU and GPU.

For this assignment, we have provided the code for dataset creation from a single image. The input image is provided in the folder and can also be accessed by the code for dataset creation.

Questions

1. **Import a multimodal network that can be used to do object detection.**
   a. **Which network did you pick and why?**
   b. **How are you passing the image to this network? You might want to consider the different ways of cropping the images into patches.**
2. **Which metric are you using for similarity between the text prompt and the image?**
3. **Is the method you are using different from YOLO? Elaborate your answer. If yes, what would be the advantages and disadvantages over YOLO.**

Optional but fun!

4. **Can you think of a method to adapt the architecture to give an end to end framework? i.e. given the image and prompt, the output is a cropped image or the original image with a bounding box drawn.**

ANSWERS:-

1. a) I picked the CLIP multimodal network by OpenAI because that is the multimodal network we recently learned and it is trained on 400 million image-text pairs and is very good with text and image encoding.

b) The image is broken into patches of size similar to the sample output displayed and for the network to work better and to demonstrate its effectiveness, I have increased the number of patches by introducing overlap between the different patches.

2. The metric I am using to compute the similarity between the text prompt and the image is the cosine similarity. It is readily available in the PyTorch package and is one of the most commonly used similarity metrics between text and images.

3. The method I am using is different from YOLO because YOLO involves one network where box regression and classification is done at the same time end to end and does not involve a region proposal generator like previous two-stage detectors. My method involves a very

passive regional proposal generator where I am just splitting the image into different patches and I am checking if the prompt is similar to the patch. YOLO detects objects on the go and box regression takes place and there are only 49 classes that can be predicted whereas my network has a lot of generality and can predict any class based on the text prompt received. YOLO does not take in a text prompt and only predicts the classes it was trained on. My network finds the best patch among all the patches of the image which is the most similar to the text prompt and gives that as the output whereas YOLO takes in an image as a whole and finds objects in it without the use of patches or region proposal generators.

My network is a good tool for zero knowledge tutorials where the cropped image of the object is given as an output whereas YOLO gives us a bounding box where we are able to locate the object within an image. Our approach just outputs the cropped image but we have no idea where they are and it is zero knowledge because we know the object is there but we do not know anything else like where it is located.