Homework Assignment 1: Text-based image retrieval

CSE 597-004, Fall 2022

September 20, 2022

1 Introduction

For this assignment, we are going to focus on the implementation of Interactive Text-based Image Retrieval (ITIR). The difference between ITIR and Text-based Image Retrieval (TIR) is that TIR model takes the text **T** as input and try to find the best image **I** that matches the description **T**. However, in ITIR, the model takes one candidate image I_c and the text which describes how the image I_c should change as inputs, and outputs the target image I_t. Nothing changes much.

2 Setups

For this and the rest of the assignments, we are going to use Python and Pytorch. We provide the tutorial of Python and Pytorch for those who are not so familiar with them.

We suggest students complete the networks in Google Colab. If you'd like to complete the assignments in colab, you can visit the colab website and upload the notebook. To use a GPU, set your runtime to include a hardware accelerator. Students may also complete the homework locally with Jupyter, though training your network will be fairly slow on a CPU. The following are the steps to get the code and data ready.

0. Get the code from Canvas and unzip. If you are using Colab, you could upload each file one by one to Colab.
1. Download the data from

   https://drive.google.com/file/d/18lDDTXGxiw6JFLf6b9j4kQIaZsyJOBQP/view?usp=sharing

   and fully unzip the them (there are three zip files in zipped in one file)
2. Put the data/ folder under ITIR/. On Colab, you can upload the data to your Google Drive and mount the data. Remember to change the path to data accordingly.
3. Resize the images.

   ```
   python resize_images.py
   ```

   If you use Colab and you have a pro account, you can just use the terminal provided by Colab. If you do not have a pro account, you can just type !bash in Colab and a bash will appear where you could run your command.

   ```
   !bash
   ```

4. Build vocabulary for dress. We only focus on dress images for this assignment. After this step, you would expect *dict.dress.json* under data/captions

```
python build_vocab.py --data_set dress
```

3 TODOs

You are expected to implement the following parts of the pipeline:

(1) Generic-Feature Extraction (2.5 points)

(2) Common Space Embedding (2.5 points)

(3) Similarity Measurement (2.5 points)

(4) Triplet ranking loss (2.5 points)

You will fill in the missing codes in models.py, utils.py, and train.ipynb to complete the task where TODO is clearly marked (there are **6** TODO blocks). The expected best dev score will be around from 0.75 0.80.

4 Submit

For submitting the assignment, simply upload the completed **train.ipynb** file. Be sure the cells have output from running your code. You do not need to include any other files (checkpoints, images, or h5py).