COS 529 Assignment 4: Visual Question Answering

Due 11:59pm ET 12/06/2022

Collaboration policy This assignment must be completed individually. No collaboration is allowed.

1 Visual Question Answering:

Visual question answering combines image understanding with natural language processing. In this assignment you will be given the following input: an image and a question. The task is to answer the question using the information provided by the image. Several examples of image/question pairs from the VQA set are shown in Figure 1.

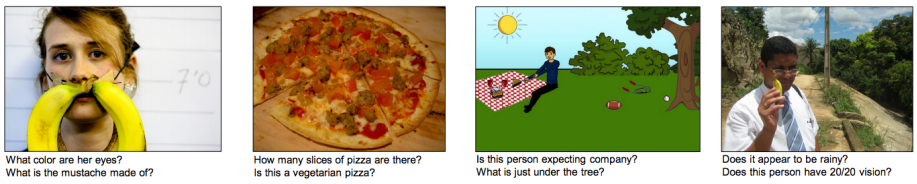

Figure 1: Sample image question pairs. Source: https://visualqa.org/

In this assignment, you will be working with the *Balanced Real Images*. The training set contains about 80k images from coco along with 400k questions and answers. For this assignment, you will be training a model to perform VQA, and will be evaluated based on your performance on the validation set.

2 Task

Start by downloading the training and validation images from the downloads page: https://visualqa.org/download.html. If you go to the downloads page you will find 3 separate datasets (Balanced Real Images, Balanced Binary Abstract Scenes, and Abstract Scenes). We will be working the Balanced Real

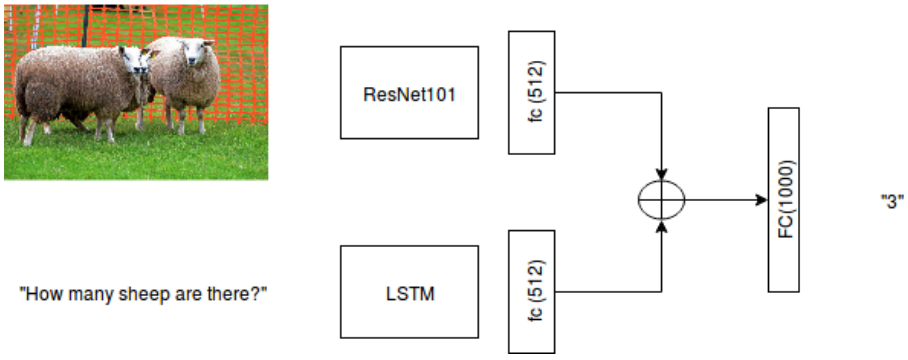

Figure 2: Baseline VQA Model

Images on this assignment. Download training/validation questions and the training/validation answers. If you do not have access to a GPU, we recommend using pretained features extracted using Resnet101. If you want to train your own image feature extractor, you can also download the training and validation images from the VQA download page. The baselines in this assignment were established by using these features provided. However, you are free to train your own image encoder.

In this assignment, you will need a way to extract features from the input questions. One option is to use an LSTM as we discussed in class, which can be trained using word embeddings as input. If you are unsure as to how to start, we recommend taking a look at the paper which introduced to dataset: https://arxiv.org/pdf/1505.00468.pdf.

3 Baseline

We implemented as baseline design shown in Figure 2. The baseline is composed of two encoders: (1) the image encoder maps each image to a feature vector and (2) the question encoder maps each question to a feature vector. The image encoder applies a fully connected layer to the features from a pretrained Resnet101 model. The question encoder uses a word embedding and a 2 layer LSTM. We then perform element-wise addition between the image vector and the output of the LSTM. We then apply a final fully connected layer with softmax activation. VQA is open-ended so there are a massive space of possible answers. We choose the most common 1000 answers and train the model using cross-entropy loss.

Our baseline model achieves a performance of 47.7% on the validation set. In order to receive full credit on this assignment, you must a achieve an accuracy of at least 45%.

4 Task

In this assignment, you will be designing your own VQA system. You can train a full model jointly (image encoder and question encoder) or you may use the features we have provided. For evaluation, you need to use the evaluation code found here: https://github.com/GT-Vision-Lab/VQA, which includes an evaluation demo and a sample results file format. You will need to modify the paths in the `vqaEvalDemo.py` so that they point to the correct locations of the validation annotations, questions, and your results.

You can download pre-extracted features for the training and validation images by using the following links:

- Training: `https://www.dropbox.com/s/r56cyszpi7dpokn/train.pickle?dl=0`
- Validation: `https://www.dropbox.com/s/rubxestevay06y7/val.pickle?dl=0`

These are python pickle files storing dictionaries which contain the extracted weights. The weights can be accessed by indexing the dictionary with the image_id. For example, if you want the features for image_id=12.

```
coco_train = pickle.load(open("coco_train.pickle", "rb"))
feats = coco_train[12]
```

Restrictions: You may use deep learning libraries such as PyTorch and Tensorflow for this assignment. However, you many not use any existing code which has been written for VQA. For example, if you find submission code for the VQA challenge on github, you may not use this code in your implementation.

You will turn in a 4-page report describing your method. In addition, please submit your code as well for this assignment. You may include whatever information you feel is important, but your report must contain the following:

1. **Description of your approach.** (a) How did you extract features from the question? (b) How did you extract features from the image? (using the provided features is fine) (c) How did you combine image and question features? (d) How did you use the combined features to select an answer to the question? (e) How as your model trained?
2. **Accuracy.** What was the accuracy of the VQA system?
3. **Examples.** Include some question/image pairs from the *validation* set and your model's answers to the question.
4. **Ablation:** We ask that you perform 1 informative ablation experiment where you make a meaningful change to your model, and measure how making this change impacts the performance of your method. One example may be replacing an LSTM with a bag-of-words for question encoding. You can also try and test the accuracy of your model if you use only the question as input, or only the image as input.

5. **Reflection.** (a) Name at least one thing you tried that didn't initially work in your system. What did you do to get it to work?

6. **Next steps.** (a) What would be the next steps you would try to further improve the accuracy? (b) How much of an improvement do you get, or think you could get? You can also experiment directly with next steps; and providing results and careful interpretation will be sufficient.

We will be grading your report on the quality of your answers in addition to the performance of you model, so make sure your answers are thorough. We are primarily interested in high level discussion regarding the overall motivation for your design.