Image Captioning Using Gated Recurrent Units

Image captioning is the task of generating a sequence of words (caption) that describes the content of an image. Image captioning helps improve content accessibility for people by describing images to them. In this assignment, you are required to build a deep learning model that can learn to generate a caption for a given image.

Dataset:

For the image captioning task, you will use the Flickr8k dataset. This dataset comprises over 8,000 images, each paired with five different valid captions. The dataset has two compressed folders:

- ***Flickr8k_Dataset.zip*** which contains the images.
- ***Flickr8k_Text.zip*** which contains some metadata. You should only use the "***Flickr8k.token.txt***" from this folder as this file contains the names and corresponding captions of the images.

You can download and unzip the dataset manually from the hyperlink above, or you can use the following commands in your notebook to obtain it:

```
!wget -q https://github.com/jbrownlee/Datasets/releases/download/Flickr8k/Flickr8k_Dataset.zip
!wget -q https://github.com/jbrownlee/Datasets/releases/download/Flickr8k/Flickr8k_text.zip
!unzip -qq Flickr8k_Dataset.zip
!unzip -qq Flickr8k_text.zip
!rm Flickr8k_Dataset.zip Flickr8k_text.zip
```

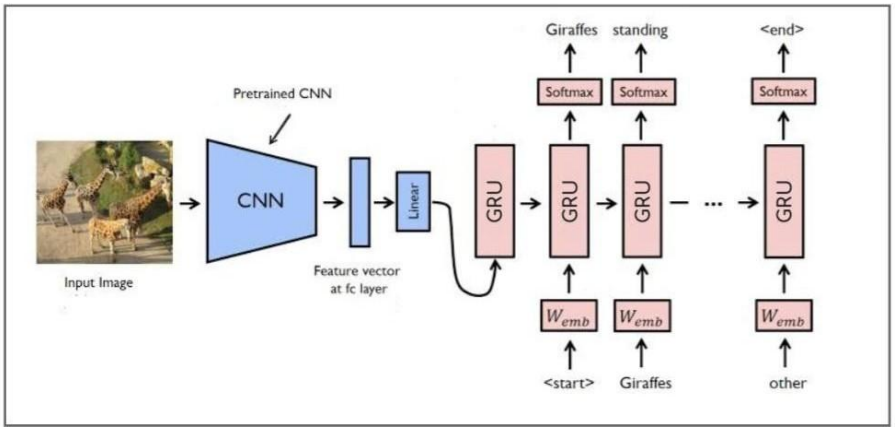
Note: You can use the entire dataset or subsample it.

Requirements:

In a notebook, write Python code where you:

1. Load the raw images and their captions.
2. Perform the necessary ***image preprocessing***.
3. Perform the necessary ***text preprocessing*** on the captions and vectorize them.
   (You will be asked which preprocessing steps you performed and why)
4. Use a ***pre-trained CNN*** model to extract the features of each image.
5. Prepare the dataset (images' feature vectors and caption vectors) that will be used by the caption generation model, and split the dataset into train and test sets.
6. Build your own ***img2seq GRU model***. The model should consist of an encoder and a decoder. An example of the model is illustrated in the figure at the bottom of the page.

   Note: You are allowed to use a masking layer, an embedding layer, and multiple GRU layers if you see fit.
7. Train the model on the train set and evaluate it on the test set.



8. Use the trained model (in a function that performs the inference step) to generate a caption for a new image. Take five different images of your surroundings using your mobile's camera, send them to the inference function to generate their captions, and display the results (images with captions).