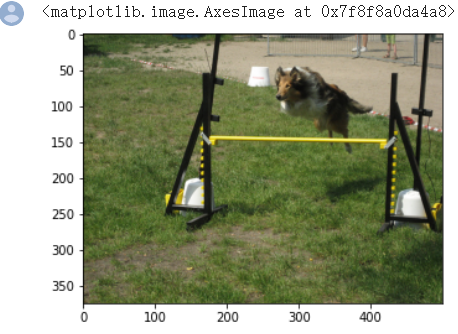
**COMP5623 Coursework on Image Caption Generation**

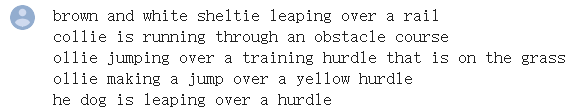
Fanhui Meng (sc19fm) 201373470

1. Lemmatize means to restore a language vocabulary of any form to a general form (can express complete semantics), which means people can use fewer words to express the meaning of a sentence. So the sentence made of lemmatise words can be short. It can be much more efficient and useful when applying this method to information retrieval and text and natural language process. As for the regular sentence, it may contain a lot of meanless words, and it may cause inefficiency during the natural language process.

However, the process of lemmatising words can be complicated and hard to implement. It needs to return the original form of the word, analyse the form of the word, etc. No one can guarantee those processes are accurate and perfect. Especially for those complicated processes, which are more likely to get wrong and cause imprecise during the natural language process.

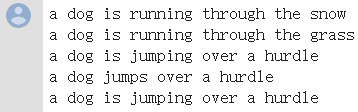
1. Below are the first image and its original captions.



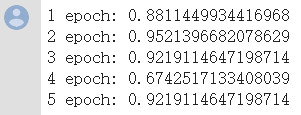


Here are the generated captions before training

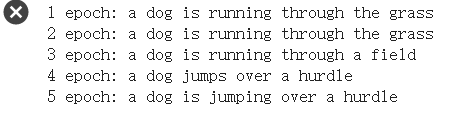
Here are the generated captions after each epoch of training for RNN decoder:



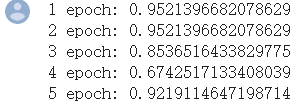
And their BLEU scores for RNN decoder:



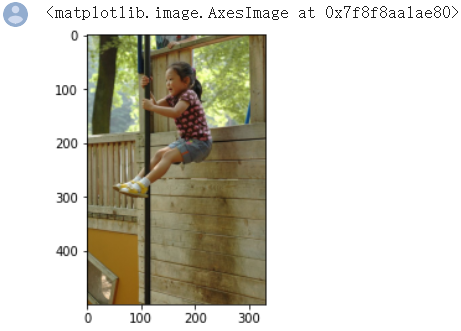
Here are the generated captions after each epoch of training for LSTM decoder:

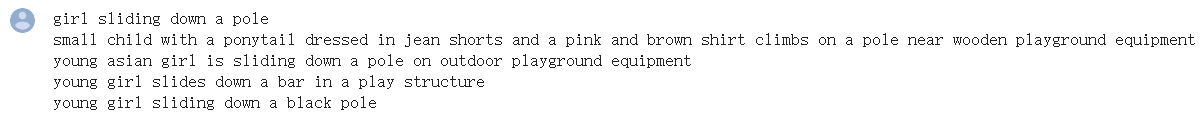


And their BLEU scores for LSTM decoder:

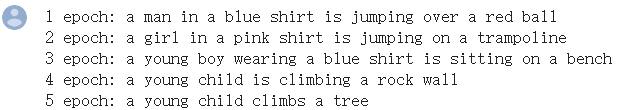


Below are the second image and its original captions.

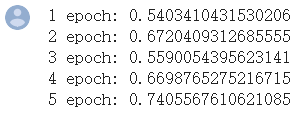




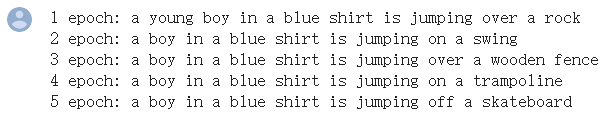
Here are the generated captions after each epoch of training for RNN decoder:



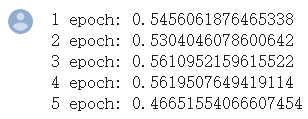
And their BLEU scores for RNN decoder:



Here are the generated captions after each epoch of training for LSTM decoder:

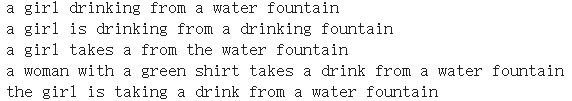
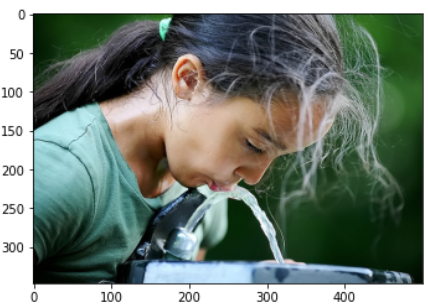


And their BLEU scores for LSTM decoder:



Five additional test image for RNN:

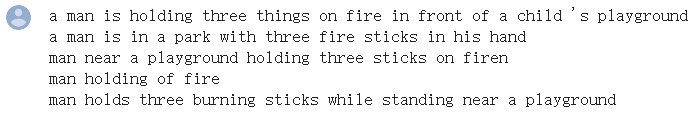
(The order from top to bottom is the original image, 5 references captions, generated captions and the BLEU score)





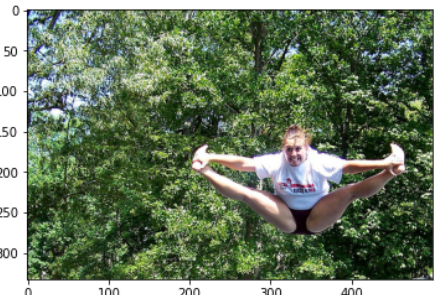


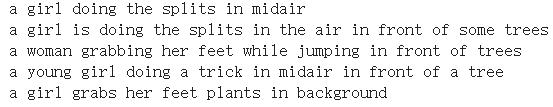








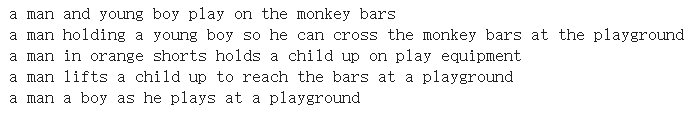






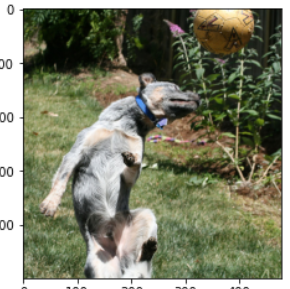


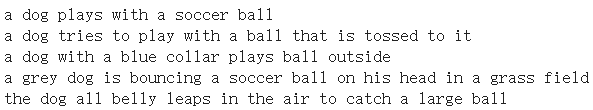










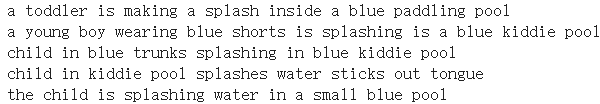






Five additional test image for LSTM:

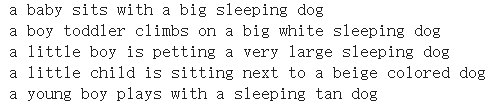








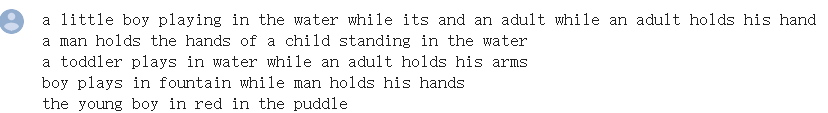








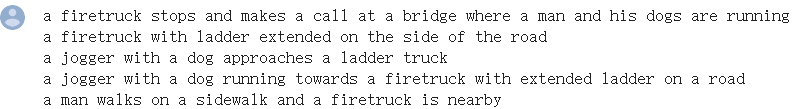






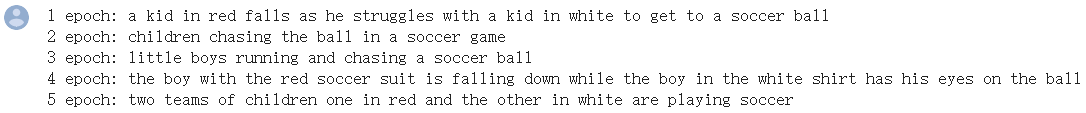


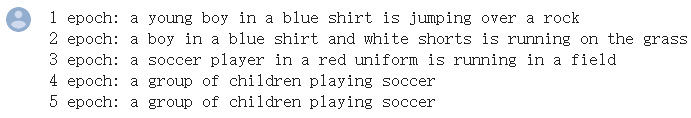


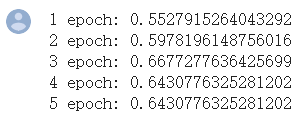












1. As for loss, I copy these losses into two txt files, so that I can compare them easily.

In the beginning, both RNN and LSTM start with about 8.1 loss. After the first three epoch training, the losses are quite close, but the RNN network seems a lower loss.

(Left is LSTM, and the right-hand side is RNN)







And after epoch 4 and 5 training, the LSTM has a lower loss.





In conclusion, the loss of RNN training would drop faster in the first three epoch. After this, the LSTM loss would be lower. And the trend of LSTM loss would be more stable, and it goes around 2.4 and 2.2 in the final epoch. The RNN loss would be more fluctuant, and it goes around 2.6 and 2.2 in the last epoch.

I pick 2 picture in the test set and test their generated captions and their score.

The first picture has five short sentences, and both RNN and LSTM have good score in this case.

The best score of RNN is approximate 0.95214, and the average is 0.870273

The best score of LSTM is approximate 0.95214, and the average is 0.870818

For the picture with apparent features and original short sentences. Both RNN and LSTM can have a good score in this case. LSTM seems has a better score according to the average score, but their scores are close.

In terms of the quality of generated captions. Both RNN and LSTM captions describe the feature of the image correctly. However, the first sentence generated by RNN recognise the white object wrongly as snow. And the LSTM bare make mistakes. In conclusion, the qualities of RNN and LSTM are both excellent and closed, but the LSTM has a little bit better performance in this case.

The second picture that I pick from the test set as well as the original five sentences are more complicated than the first one.

The best score of RNN is approximate 0.74056, and the average is 0.63636

The best score of LSTM is approximate 0.56195, and the average is 0.53311

It's quite clear that both RNN and LSTM have a much lower score than the first simple image. This is because the image is more complicated and have more features. And it’s generated captions are harder to match the longer reference captions. Also, in this case, RNN has a better score than LSTM, which is strange to me. Theoretically, RNN has trouble with long term memory due to the possibility that its gradient is unusually small or large. And LSTM can prevent gradient disappearance and has long term memory. So LSTM should perform better than RNN in long caption case.

When I tried another different 5 images, it turns out LSTM can have better performance, which means in most of the cases, LSTM can get higher score than RNN.

In the aspect of the trend of BLEU score. The LSTM score tends to be more stable, which means the best and worst scores are close. And RNN scores are unstable. It's the best score can be very high, and the worst is pretty low.

In the aspect of generated sentences quality. LSTM can generate long sentence to describe one image. And the RNN generated sentences are quite short, which means it contains less information, and sometimes it will get a low score due to lack of information.

1. One picture can have different descriptions. Some of them can be simple and short; others can be long and detailed. They are both right, but people want the generated sentences are as fully detailed as possible. Otherwise, some models can always generate some indistinct sentence and get a good score.

In the *ExpertAnnotations.txt*, it comes up with a ranking of each sentence. When evaluating the model, we can put weight on each sentence. For example, if the generated captions match the most with the sentence with score 4, then this model performed exceptionally well. If the generated captions only match the sentence with score 1, then we think this model is not good enough.

In the *CrowdFlowerAnnotations.txt* file, has also created a ranking system. The sentence with the most yeses from the judgments must be the most valid. Then put more weight on the most valid sentence when evaluating the model.

1. Please see the code in *COMP5623\_CW2\_Fanhui.ipynb* file.