COL774 Assignment 4

Mustafa Chasmai | Tamajit Banerjee
2019CS10341 | 2019CS10408

November 2021

1 Non Competitive Part

1.1 Encoder

We used the general ResNet 50 architecture for our encoder. ResNet 50 is a convolutional neural network that is 50 layers deep, with multiple convolutional and residual blocks added alternatingly. We implement a generic resnet architecture that works for some input image, and generates a feature vector of a given dimension. For the encoder, we use an embedding dimension of 128.

1.2 Decoder

We use the LSTM module provided by pytorch directly for the decoder. We extract embeddings from the captions, concatenate them with the features of the image given by our encoder and the embeddings of the caption, and pass this concatenated tensor through the LSTM module. Finally, we apply a fully connected layer on the LSTM outputs and use this as the predictions for the decoder.

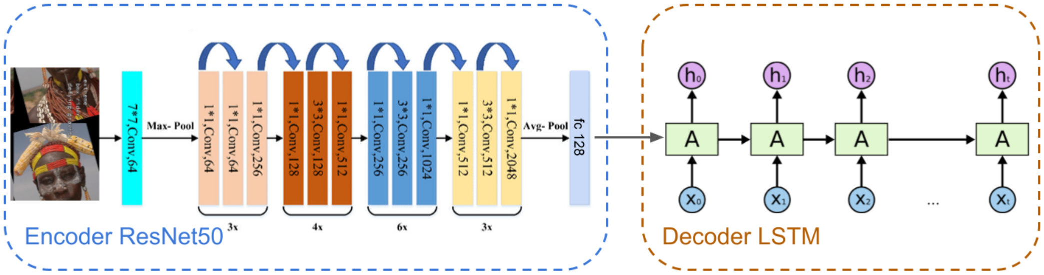

Figure 1: The Encoder decoder architecture

1.3 Training and Inference

We trained the architecture with cross entropy loss, using Adam optimiser for 10 epochs. We used normalisation and pad-resize transforms for the image data and vocabulary related transforms for the captions. The variation of loss can be seen in the figure below.

For Inference, we used the beam search method described, with a beam size of 3 and maximum length of 8. We obtained a BLEU score of 0.1390 on the test set.

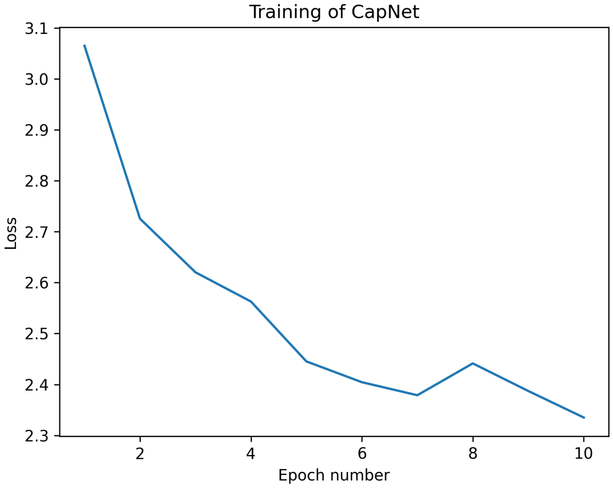

Figure 2: Training of CapNet

2 Competitive Part

For the Competitive part, we replaced our encoder with a pre-trained Resent-50 model available from torchvision. This was pre-trained on ImageNet, a large scale image classification dataset. The last few layers were removed to obtain the features instead of classification outputs. The deocoder LSTM as well as embedding was kept the same, and was learnt from scratch.