A Basic Encoder-Decoder Model

Amitai Yacobi

June 3, 2022

Architecture, Loss function and Optimizer

As for the architecture, I used unidirectional LSTM network for both Encoder and Decoder. After some experiments with the network dimensions I found that the best results were for the following dimensions:

1. Embedding dimension of 128 for both Encoder and Decoder.
2. Hidden dimension of 256 for both Encoder and Decoder (which means that the input dimension for the Decoder's LSTM after concatenation is the hidden dimension + the embedding dimenstion which is $128 + 256 = 384$).

As for the loss function, I used the Negative Log Likelihood loss like we saw in the lectures. In practice, I summed the negative log likelihoods for each correct output token in the output sequence, as computed by the softmax function.
As for the optimizer, I tried different types of optimizers like RMSprop, Adam and AdaDelta and I found that Adam gave best results.

Hyper parameters

After some tuning of the hyper parameters I found that the following values gave me the best results:
Learning rate - 0.0005 for both Encoder and Decoder
Dropout - 0.1 for both Encoder and Decoder

Results

Loss on the train set - 0.75
Loss on the dev set - 5.11
Bleu score on the dev set - 80.32
Bleu score on the test set - 82.244

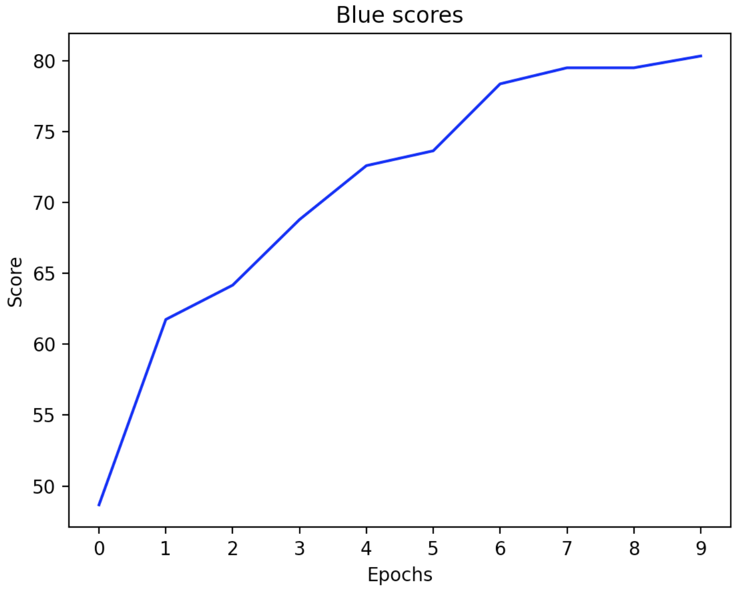

Figure 1: Bleu score on dev per epoch

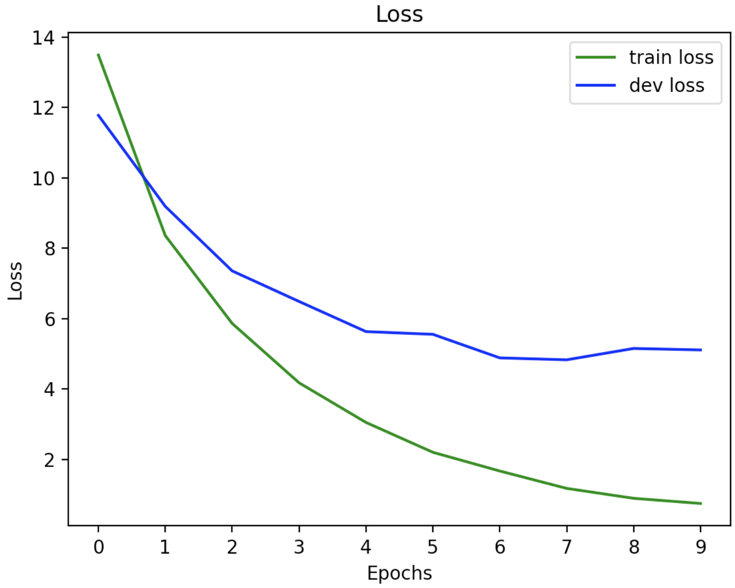

Figure 2: Loss on train and dev per epoch