11-731 Machine Translation

Assignment 1 Report

Kaiyu Zheng

kaiyuz@andrew.cmu.edu

1 Overview

In the first assignment, I implement the machine translation model based on the idea from the paper and reference in Deep Learning course. After following the basic idea from these resource, I do some experiments to tune the model and pass the baseline with the final model.

2 Implementation

2.1 Sentences

Given sentences with different length, I process them with padding, transform from word to index, masking and embedding before putting them to the model. These operations should be done both for encoder and decoder for the model.

2.2 Model architecture

For the model architecture, the encoder is one layer bidirectional LSTM, which solves the vanishing gradient problem and the long distance between words by reducing the length of dependencies. In the decoder part, the initial LSTM cell is valued by output from the encoder and the input of the cell is based on the idea of input-feeding approach by concatenating the attentional vectors with input. The teacher forcing is used in each iteration, which sometimes uses output from prior steps as input. The calculation of the attention score is simply based on the dot product and the mask. For the translation, instead of beam search, I just implement the greedy search.

3 Experiments

3.1 Hyper-parameter

Parameters are set with same values as default in the script without much experiment.

Batch size	hidden size	embed size	uniform init	dropout	clip grad	lr decay
64	256	256	0.1	0.2	5.0	0.5

3.2 Basic model

Following the idea of the model mentioned in the last part, the basic model get following BLEU score in test and valid set. Unfortunately, the result is not good enough.

Test: BLEU = 17.68, 50.0/23.5/12.4/6.7 (BP=1.000, ratio=1.031, hyp$_len =$ $135219, ref_len = 131141$)

Valid: BLEU = 18.82, 51.1/25.0/13.4/7.3 (BP=1.000, ratio=1.033, hyp$_len =$ $133360, ref_len = 129091$)

3.3 Uniform initialization

Following the idea from the piazza, I apply the uniform initialization [-0.1, +0.1] for all parameters, which greatly increase the result shown as below. By searching this idea online, I guess the idea of this trick is to somehow make the parameter be learned better in the process of back propagation.

Test: BLEU = 24.68, 58.9/31.8/18.9/11.4 (BP=0.979, ratio=0.979, hyp$_len =$ $128365, ref_len = 131141$)

Valid: BLEU = 26.53, 61.1/34.2/20.7/12.9 (BP=0.970, ratio=0.971, hyp$_len =$ $125323, ref_len = 129091$)

3.4 Dropout

The last try is to add dropout in attention vector. I add one more layer to the read out layer and do the dropout in this time. It just change the result a little.

Test BLEU = 24.72, 62.0/33.8/20.1/12.2 (BP=0.923, ratio=0.926, hyp$_len =$ $121430, ref_len = 131141$)

Valid BLEU = 26.73, 63.7/36.2/22.1/13.7 (BP=0.924, ratio=0.927, hyp$_len =$ $119673, ref_len = 129091$)

4 To do list

The strategy for preprocessing sentences would be a good option, such as unknown words replacement and the threshold of rare words. These operations should be done both for encoder and decoder for the model. The strategy for preprocessing sentences would be a good option, such as unknown words replacement and the threshold of rare words. These operations should be done both for encoder and decoder for the model.