CS224N Assignment

Meng Yan

2021/8/12

1 Neural Machine Translation with RNNs

In Machine Translation, our goal is to convert a sentence from the source language (e.g. Cherokee) to the target language (e.g. English). In this assignment, we will implement a sequence-to-sequence (Seq2Seq) network with attention, to build a Neural Machine Translation (NMT) system. In this section, we describe the training procedure for the proposed NMT system, which uses a Bidirectional LSTM Encoder and a Unidirectional LSTM Decoder.

(g) (3 points) (written) The generate sent masks() function in nmt model.py produces a tensor called enc masks. It has shape (batch size, max source sentence length) and contains 1s in positions corresponding to 'pad' tokens in the input, and 0s for non-pad tokens. Look at how the masks are used during the attention computation in the step() function. First explain (in around three sentences) what effect the masks have on the entire attention computation. Then explain (in one or two sentences) why it is necessary to use the masks in this way.

Answer The mask in the step() function sets the padded location in the attention vector e_t to the negative infinity. Therefore, the effect of the mask is to make the probability of "pad" in the attention vector to be zero. Without the mask operation, the decode will still use the information of "pad" of hidden states, which is out of the expectation.

(h)Please report the model's corpus BLEU Score. It should be larger than 10.

Answer Run on google colab gpu, 15 mins, It early stopped at epoch 7. BLEU: 12.36

```
/content/drive/My Drive/a4# sh run.sh test
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
load test source sentences from [./chr_en_data/test.chr]
load test target sentences from [./chr_en_data/test.en]
load model from model.bin
Decoding:   0% 0/1000 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-pa
To keep the current behavior, use torch.div(a, b, rounding_mode='trunc
  return torch.floor_divide(self, other)
Decoding: 100% 1000/1000 [00:49<00:00, 20.02it/s]
Corpus BLEU: 12.362374501822375
```

(i)(4 points) (written) In class, we learned about dot product attention, multiplicative attention, and additive attention. As a reminder, dot product attention is $e_{t,i} = s_t^T * h_i$, multiplicative attention is e_t,$i = s_t^T W h_i$, and additive attention is $e_{t,i} = v^T tanh(W_1 h_i + W_2 s_t)$.

i. (2 points) Explain one advantage and one disadvantage of dot product attention compared to multiplicative attention.

Answer: 1. advantage: dot multiplication is easy to calculate, without the additional W variable. 2. disadvantage: the result of the dot multiplication is scalar, contains less information.

ii. (2 points) Explain one advantage and one disadvantage of additive attention compared to multiplicative attention.

Answer: 1. advantage: additive attention has a similar complexity as the multiplicative attention, but fits more complex situations in the experiments. 2. disadvantage: additive attention needs more hyperparameters.

2 Analyzing NMT Systems

(a) (2 points) In part 1, we modeled our NMT problem at a subword-level. That is, given a sentence in the source language, we looked up subword components from an embeddings matrix. Alternatively, we could have modeled the NMT problem at the word-level, by looking up whole words from the embeddings matrix. Why might it be important to model our Cherokee-to-English NMT problem at the subword-level vs. the whole word-level? (Hint: Cherokee is a polysynthetic language.)

Answer **:**1. It can generate embeddings for unknown words. Cherokee vocabulary is not large, and also this language is polysynthetic, so using sub-words can help to generate new word embeddings. 2. It solves out of vocabulary (OOV) problem.

(b) (2 points) Character-level and subword embeddings are often smaller than whole word embeddings. In 1-2 sentences, explain one reason why this might be the case.

Answer: The embedding size for character-level embedding is lower than the word-level embedding. It indicates that lower dimensions can represent each character. In addition, the size of character-level vocabulary is smaller than the word-level.

(c) (2 points) One challenge of training successful NMT models is lack of language data, particularly for resource-scarce languages like Cherokee. One way of addressing this challenge is with multilingual training, where we train our NMT on multiple languages (including Cherokee). You can read more about multilingual training here. How does multilingual training help in improving NMT performance with low-resource languages?

Answer Multilingual training processes multiple languages using a single translation model. This leads to transfer study and helps to gain insights through training on one language, which can also be applied to the translation of others. Multilingual models learn shared representations for linguistically similar languages without the need for external constraints, validating long-standing intuitions and empirical results that exploit these similarities. Thus, it helps to boost the performance with low-resource languages.

(d) (6 points) Here we present three examples of errors we found in the outputs of our NMT model (which is the same as the one you just trained). The errors are underlined in the NMT translation sentence. For each example of a source sentence, reference (i.e., 'gold') English translation, and NMT (i.e., 'model') English translation, please: 1. Provide possible reason(s) why the model may have made the error (either due to a specific linguistic construct or a specific model limitation). 2. Describe one possible way we might alter the NMT system to fix the observed error. There are more than one possible fixes for an error. For example, it could be tweaking the size of the hidden layers or changing the attention mechanism.

Answer

(1) 1. reason: missing "a crown of daisies" phrases. It is probably caused by the lack of similar phrases related to "a crown of daisies" in the corpus.

2. possible way: train the model with the vocabulary containing similar phrases.

(2) 1. reason: "she" is translated into "it's". bias in training sample data that doesn't have enough training sample data pairs.

2. possible way: include more gender unbiased data into the training set.

(3) 1. reason: "Littlefish" here is a specific proper noun but the translation model translates into "little fish". The reason may be the lack of corresponding pairs.

2. possible way: train the models on the vocabulary containing corresponding language pairs.

(e) Now it is time to explore the outputs of the model that you have trained! The test-set translations your model produced in question 1-i should be located in outputs/test outputs.txt.

i. (2 points) Find a line where the predicted translation is correct for a long (4 or 5 word) sequence of words. Check the training target file (English); does the training file contain that string (almost) verbatim? If so or if not, what does this say about what the MT system learned to do?

Answer:

predicted sentence: And as he took the bread, and he had given thanks, he took it, and gave to them, and said, Let us; this is my body.

target: And as they were eating, he took bread, and when he had blessed, he brake it, and gave to them, and said, Take ye: this is my body.

"this is my body" is contained in the training file. MT system learn to generate the word in this sentence one by one.

ii. (2 points) Find a line where the predicted translation starts off correct for a long (4 or 5 word) sequence of words, but then diverges (where the latter part of the sentence seems totally unrelated). What does this say about the model's decoding behavior?

predicted translation: "And they were all amazed, and they were amazed, and said one to another, Yes not all these things?"

target sentence: "And they were all amazed and marvelled, saying, Behold, are not all these that speak Galilæans?

This means the decoding behavior ignores the context of the translation sentence, therefore, generate unrelated ones.

(f) BLEU Score:

(i) 1. c_1: $p_1 = (0+1+1+1+0)/5 = 0.6$

$p_2 = (0+1+1+0)/4 = 0.5$

$len(c_1) = 5, len(r) = 4, BP = 1$

$BLEU = 1 * exp(\lambda_1 * \log p_1 + \lambda_2 * \log p_2) = exp(0.5 * log(0.6) + 0.5 * log(0.5)) = 0.5477$

2. c_2: $p_1 = (1+1+0+1+1)/5 = 0.8$

$p_2 = (1+0+0+1)/4 = 0.5$

$len(c_2) = 5, len(r_2) = 4, BP = 1$

$BLEU = exp(\lambda_1 * \log p_1 + \lambda_2 * \log p_2) = exp(0.5 * \log(0.8) + 0.5 * \log(0.5)) = 0.6324$

The second is a better translation according to the BLEU score. And I agree with the score.

(ii) For c_1: $p_1 = (0+1+1+1+0)/5 = 0.6$

$p_2 = (0+1+1+0)/4 = 0.5$

$len(c_1) = 5, len(r_1) = 6. BP = exp(1-6/5) = 0.8187$

$BLEU = 0.8187 * exp(0.5 * log(0.6) + 0.5 * log(0.5)) = 0.45$

For c_2: $p_1 = (1+1+0+0+0)/5 = 0.4$ $p_2 = (1+0+0+0)/4 = 0.25$

$len(c_2) = 5, len(r) = 6, BP = exp(1-6/5) = 0.8187$

$BLEU = 0.8187 * exp(0.5 * log(0.4) + 0.5 * log(0.25)) = 0.2589$

The first translation is better according to the BLEU. And I don't agree with it.

(iii) The good translation may receive low BLEU score, due to the little n-gram overlaps in a single reference translation. By providing multiple possible reference translation, more n-grams are available for the model and to receive a good BLEU score.

(iv) Adv: 1. More efficient than human, fully automated. 2. Cheap to implement. Human resources are expensive and need time to understand the translations.

Disadv: 1. It's about quantity measurement, not about quality of the sentence. 2. It's only calculate the occurrences of the n-grams, without checking the grammar mistakes, or the meanings.