CS 224N - Assignment 4

Ilya Kachan

July 14, 2021

1 Neural Machine Translation with RNNs

(g) Attention scores $\mathbf{e}_t$ are calculated on all encoder hidden states regardless of what words in the original sentences they correspond to: real words or padding symbols. Encoding masks mark out the positions of padding symbols and replace the corresponding attention scores with $-\infty$. As result, the attention weights $\mathbf{a}_t$ that correspond to padding symbols are zeros (after applying softmax). Therefore, using the masks helps not to attend to dummy word and attend only to the real sentence words while computing next decoder hidden state.

(h) Model's corpus BLEU score is 12.62137.

(i) i. Dot product attention is less flexible than multiplicative attention since it has no trainable parameters. At the same time it's more cheep to compute and decreases the total number of model parameters.

ii. Multiplicative attention is faster and more space-efficient that additive attention. While additive attention has more params and non-linearity that in theory might help to capture more complex patterns.

2 Analyzing NMT Systems

(a) For Cherokee to English translation it's improtant to do modelling at the subword level since Cherokee is a polysynthetic language. In Cherokee there are often quite long words that are composed of many sub-components having their own meaning. These subcomponents (morphemes) may be or may not be separated by space characters.

(b) The words themselves contain more information that separate characters, so they require more vectore space to store the information. Also there is less variaty in characters than in vords, i.e. vocabulary sizes for characters and words are dramatically different, so there is not so much need to have high-dimesionall character embeddings to express this variaty.

(c) Multilingual training is effective at generalization, and capable of capturing the representational similarity across a large body of languages. Thus for low-resource languages such training might help to learn and use the similarity between these languages and other, having more resources for training.

(d) i. In the first example we see grammatically incorrect construction. It might be caused by the limitation of the model, namely — its beam search component: the model chose the "her hair" word combination too early and tried to adapt the sentence ending to it. Increasing the beam size might help in this case, I think.

ii. In the second example we see an incorrect coreference resolution ("She" $\leftrightarrow$ "It"). This is again the limitation of the model and this issue might potentially be resolved by using self-attention mechanism (as in transformers).

iii. The third example is about a specific language construct: the model recognized the name "Littlefish" as a "small fish", i.e. behaved too literally in this case. Probably, adding POS tags to the inputs might help to minimize this issue.

(e) i. A line "I dont want to die." was translated fully correctly. Note that exactly this line was present in the training data. Thus, MT system learned to restore even some complex translations it has already seen.

ii. One example of quite a long sentence where the translation and reference prefixes match:

Reference: And when they heard these things, they held their peace, and glorified God, saying, Then to the Gentiles also hath God granted repentance unto life.

Translation: And when they heard these things, they were filled with God, and glorified God, saying, This is the God of God according to the life of life.

This says about the decoder behavior that it chose the incorrect hypothesis during the beam search and tried to complete it as it could. The incorrect choice might happen because of model's poor ability to remember long contexts.

(f) i. First compute the lengths of translations $len(r_1) = 6$, $len(r_2) = 4$,

$len(c_1) = 5$, $len(c_2) = 5$. Now let's compute BLEU score:

$$BLEU = BP \times \exp\left(\sum_{n=1}^{4} \lambda_n \log p_n\right)$$

First of all, count 1- and 2-grams of c_1 and c_2 in r_1, r_2:

For c_1 :

n-gram	c_1	r_1	r_2
the	1	0	0
love	1	1	1
can	1	1	0
always	1	1	0
do	1	0	0
the love	1	0	0
love can	1	1	0
can always	1	1	0
always do	1	0	0

$For c_2$:

n-gram	c_2	r_1	r_2
love	1	1	1
can	1	1	0
make	1	0	0
anything	1	0	1
possible	1	0	1
love can	1	1	0
can make	1	0	0
make anything	1	0	0
anything possible	1	0	1

Let's compute BLEU for c_1. Here, we have $len(r) = len(r_2) = 4$, $BP(c_1) = 1$ and

$$p_1 = \frac{0+1+1+1+0}{5} = \frac{3}{5}, \quad p_2 = \frac{0+1+1+0}{4} = \frac{1}{2}$$

$$BLEU(c_1) = 1 \cdot \exp(0.5 \log 0.6 + 0.5 \log 0.5) \approx 0.54772.$$

Similarly compute BLEU for c_2. Here, we have $len(r) = len(r_2) = 4$, $BP(c_2) = 1$ and

$$p_1 = \frac{1+1+0+1+1}{5} = \frac{4}{5}, \quad p_2 = \frac{1+0+0+1}{4} = \frac{1}{2}$$

$$BLEU(c_2) = 1 \cdot \exp(0.5 \log 0.8 + 0.5 \log 0.5) \approx 0.63246.$$

According to the BLUE score, better NMT tranlation is c_2 = "Love can make anything possible". I do agree with that.

ii. Let's recompute the scores by excluding r_2 translation.

For c_1: $len(r) = len(r_1) = 6$, $BP(c_1) = \exp(1 - 6/5) = \exp(-0.2)$,

$$p_1 = \frac{0+1+1+1+0}{5} = \frac{3}{5}, \quad p_2 = \frac{0+1+1+0}{4} = \frac{1}{2}$$

$$BLEU(c_1) = \exp(-0.2) \cdot \exp(0.5 \log 0.6 + 0.5 \log 0.5) \approx 0.44844.$$

For c_2: $len(r) = len(r_1) = 6$, $BP(c_2) = \exp(1 - 6/5) = \exp(-0.2)$,

$$p_1 = \frac{1+1+0+0+0}{5} = \frac{2}{5}, \quad p_2 = \frac{1+0+0+0}{4} = \frac{1}{4}$$

$$BLEU(c_2) = \exp(-0.2) \cdot \exp(0.5 \log 0.4 + 0.5 \log 0.25) \approx 0.25891.$$

Now c_1 translation "the love can always do" receives the higher BLEU score. I do not agree that c_1 is better translation.

iii. One sentence may have several correct translations. Eveluating over a single reference can be problematic from two points of view: we restrit the training of our model and make is less flexible, and the evaluation results may be underestimated, since the model could produce the correct translation that does not match with the reference.

iv. BLEU's advantages over human evaliations are: human evaluation is slow and expensive, BLEU is objective criteria, while human judjements might be subjective. BLEU's disadvantages: it is purely statistical and it doesn't know almost anything about semantics, also there might be several correct translations and BLEU will penalize some that do not match with the reference.