Stanford CS 224n Assignment 4

Makanov Artem

February 7, 2023

1 Neural Machine Translation with RNNs (45 points)

In Machine Translation, our goal is to convert a sentence from the source language (e.g. Spanish) 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 (lines 295-296). 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: Where mask equals 1, the $e_{t,i}$ will equals $-inf$. Then, α_t, which we get from softmax, will be equal to zero. Corresponding hidden vectors of non-relevant word won't have any influence to the next word.

(i) (4 points) Once your model is done training (this should take about 4 hours on the VM),execute the following command to test the model: sh run.sh test. Please report the model's corpus BLEU Score. It should be larger than 10

Answer: 13.240979367048412

(j) (3 points) (written) In class, we learned about dot product attention, multiplicative attention, and additive attention. Please explain one advantage and one disadvantage of dot product attention compared to multiplicative attention. Then explain one advantage and one disadvantage of additive attention compared to multiplicative attention. As a reminder, dot product attention is $e_{t,i} = s_t^\intercal h_i$, multiplicative attention is $e_{t,i} = s_t^\intercal Whi$, and additive attention is $e_{t,i} = v^T \tanh(W_1 h_i + W_2 s_t)$

Answer:

i. Adv: When we compute scalar product of s_t and h_i, we can get the measure of similarity of these vectors. W matrix can improve this procedure in context of finding appropriate $e_{t,i}$ values for attention mechanism. DisAdv: We increase the number of trainable parameters => increase the chance of overfitting.

ii. Adv: We separate the functions of attention function into 2 parts: The first is trainable parameter v_t. The second scales the first part's components according to context of h_i and s_t. I suppose, that model can train better, if it expects that attention score values will be in some approximately the same place nearby v_t. DisAdv: We need to apply $tanh$ function for all of component of argument, which is not parallelable procedure.

2 Analyzing NMT Systems (33 points)

(a) (3 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: As Cherokee is a polysynthetic language, in which all members of the sentence are combined into a single whole. If model your NMT problem as whole word-level, we will create too big vocabulary, because there will be combinatorial explosion.

(b) (3 points) Transliteration is the representation of letters or words in the characters of another alphabet or script based on phonetic similarity. For example, the transliteration of ᏣᏅᏔᏍᎪ (which translates to "do you know") from Cherokee letters to Latin script is tsanvtasgo. In the Cherokee language, "ts-" is a common prefix in many words, but the Cherokee character Ꮳ is "tsa". Using this example, explain why when modeling our Cherokee-to-English NMT problem at the subword-level, training on transliterated Cherokee text may improve performance over training on original Cherokee characters.(Hint: A prefix is a morpheme.)

Answer: Maybe, we can improve performance because some Cherokee characters, include independent morphemes. I think it's better to split words to subwords equal to independent morphemes. For example, Cherokee character G (which transliteration is "tsa"), includes morpheme "ts-".

(c) (3 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:
https://ai.googleblog.com/2019/10/exploring-massively-multilingual.html.
How does multilingual training help in improving NMT performance with low-resource languages?

Answer: Multilingual training helps in improving NMT performance with low-resource languages because of transfer learning. Model transfers knowledge from high-resource languages to low-resource.

(d) (6 points) Here we present a series of errors we found in the outputs of our NMT model (which is the same as the one you just trained). For each example of a reference (i.e., 'gold') English translation, and NMT (i.e., 'model') English translation, please:

1. Identify the error in the NMT translation.
2. Provide possible reason(s) why the model may have made the error (either due to a specific linguistic construct or a specific model limitation).
3. 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.

Below are the translations that you should analyze as described above. Only analyze the underlined error in each sentence. Rest assured that you don't need to know Cherokee to answer these questions. You just need to know English! If, however, you would like additional color on the source sentences, feel free to use a resource like https://www.cherokeedictionary.net/ to look up words.

Source Sentence: *ᏄᏩᏁᎨᎾ ᎦᎪᏩᎳᎩᏍᎬ, ᎯᎠ ᏄᏍᏕ ᏚᏏᎳᏛ: ᏧᏓᎴᏅᏓ ᎦᎪᎡᏍᏕ ᏧᏏᎳᏛᏙᏗ ᎠᏍᏓ ᎾᏅᏂᏍᎩ.*
Reference Translation: *When she was finished ripping things out, her web looked something like this:*
NMT Translation: *When it was gone out of the web, he said the web in the web.*

Source Translation: *ᎤᏍᏗ ᎢᏉᎬᎢ, ᎦᏙᎲᎢ? ᎤᏓᏛᏛᏁᎢ ᎤᏍᏗ ᎠᏧᏣ.*
Reference Translation: *What's wrong little tree? the boy asked.*
NMT Translation: *The little little little little little tree? asked him.*

Source Sentence: *"ᎤᏓᏄᏬᏗ ᏂᎨᎡᎾ," ᎤᏛᏁ ᎨᎻ.*
Reference Translation: *" 'Humble,' " said Mr. Zuckerman*
NMT Translation: *"It's not a lot," said Mr. Zuckerman.*

Answer:

i. The error is in gender of the pronoun. The reason for the error: model has problems with saving information about gender of named entities. Possible solution: scale model to get rid of this limitation.

ii. N-grams repetitions. This error occurred because of inductive bias. We can get rid of this problem with simple heuristic – don't repeat n-grams.

iii. Semantic error. Sentiment of reference utterance is close to NMT translation, but the senses are not the same. Possible solution: scale model to get rid of this limitation.

(e) (4 points)

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?

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?

Answer:

i. 91 string "The Jews therefore said" Reference: "The Jews therefore said, Will he kill himself, that he saith, Whither I go, ye cannot come?" NMT translation: "The Jews therefore said, Are he himself? saying, Come, I go not." There are same words almost verbatim, but the sense is not correct. MT system learned to understand the meaning of the sentence and rephrase it into another form.

ii. Reference: 149 string "They answered unto him, We are Abraham's seed, and have never yet been in bondage to any man: how sayest thou, Ye shall be made free?" NMT translation: They answered and said unto him, I am Abraham's seed, and did not see us; and wherefore shall ye see me? It demonstrates that NMT system can understand general construction of sentence, but it has problems with transferring sense of it.

(f) (14 points) BLEU score is the most commonly used automatic evaluation metric for NMT systems. It is usually calculated across the entire test set, but here we will consider BLEU defined for a single example.

Suppose we have a source sentence s, a set of k reference translations $r_1, \ldots, r_k$, and a candidate translation c. To compute the BLEU score of c, we first compute the *modified n-gram precision* p_n of c, for each of $n = 1, 2, 3, 4$, where n is the n in n-gram:

$$p_n = \frac{\sum_{\text{ngram} \in c} \min\left(\max_{i=1,\ldots,k} \text{Count}_{r_i}(\text{ngram}),\ \text{Count}_c(\text{ngram}) \right)}{\sum_{\text{ngram} \in c} \text{Count}_c(\text{ngram})} \tag{1}$$

Here, for each of the n-grams that appear in the candidate translation c, we count the maximum number of times it appears in any one reference translation, capped by the number of times it appears in c (this is the numerator). We divide this by the number of n-grams in c (denominator).

Next, we compute the *brevity penalty* BP. Let $len(c)$ be the length of c and let $len(r)$ be the length of the reference translation that is closest to $len(c)$ (in the case of two equally-close reference translation lengths, choose $len(r)$ as the shorter one).

$$BP = \begin{cases} 1 & \text{if } len(c) \geq len(r) \\ \exp\left(1 - \frac{len(r)}{len(c)}\right) & \text{otherwise} \end{cases} \tag{2}$$

Lastly, the BLEU score for candidate c with respect to $r_1, \ldots, r_k$ is:

$$BLEU = BP \times \exp\left(\sum_{n=1}^{4} \lambda_n \log p_n \right) \tag{3}$$

where $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ are weights that sum to 1. The log here is natural log.

i. (5 points) Please consider this example
Source Sentence s: **ᎠᎴ ᎾᏍᎩ ᎢᎦ-ᎦᏗᏍᏗᏍᎩ ᎤᎵᏏᎬ ᏚᏎᏌᎦᎢ ᎤᎵᏏᎩᏃ ᎥᏝ ᏱᏚᏓᏂᏎᏙᎢ**
Reference Translation r_1: *the light shines in the darkness and the darkness has not overcome it*
Reference Translation r_2: *and the light shines in the darkness and the darkness did not comprehend it*
NMT Translation c_1: and the light shines in the darkness and the darkness can not comprehend
NMT Translation c_2: the light shines the darkness has not in the darkness and the trials
Please compute the BLEU scores for c_1 and c_2. Let $\lambda_i = 0.5$ for $i \in \{1, 2\}$ and $\lambda_i = 0$ for $i \in \{3, 4\}$ (**this means we ignore 3-grams and 4-grams**, i.e., don't compute p_3 or p_4). When computing BLEU scores, show your working (i.e., show your computed values for p_1, p_2, $len(c)$, $len(r)$ and BP). Note that the BLEU scores can be expressed between 0 and 1 or between 0 and 100. The code is using the 0 to 100 scale while in this question we are using the **0 to 1** scale.

Which of the two NMT translations is considered the better translation according to the BLEU Score? Do you agree that it is the better translation?

Answer:
According to the BLEU Score first translation is equally good. In my opinion, C1 is better.

cand	p1	p2	len(c)	len(r)	BP	BLEU
c1	0.846	0.750	13	13	1.000	0.796
c2	0.846	0.750	13	13	1.000	0.796

ii. (5 points) Our hard drive was corrupted and we lost Reference Translation r_2. Please recompute BLEU scores for c_1 and c_2, this time with respect to r_1 only. Which of the two NMT translations now receives the higher BLEU score? Do you agree that it is the better translation?

Answer:
Second NMT translation now receives the higher BLEU score. I disagree with this decision.

cand	BLEU
c1	0.716
c2	0.796

iii. (2 points) Due to data availability, NMT systems are often evaluated with respect to only a single reference translation. Please explain (in a few sentences) why this may be problematic. In your explanation, discuss how the BLEU score metric assesses the quality of NMT translations when there are multiple reference transitions versus a single reference translation.

Answer:
I think, when there are multiple reference translations used to compute BLEU Score, we can take into account diversity of language. In modified n-gram precision's numerator there are maximum number of n-grams in reference sentences. The more references, the more likely it is to get a higher value for better translation. In another case, we can see that good translations are underestimated.

iv. (2 points) List two advantages and two disadvantages of BLEU, compared to human evaluation, as an evaluation metric for Machine Translation.

Answer:
Adv: 1. It is faster than human evaluation. 2. It can process words form specific domain, that human may be doesn't know.

Disadv: 1. It takes into account only n-grams, so it considers the sequence of tokens as a bag of n-grams on concrete n-gram level. This prevents full understanding of context of the sentences, because of co-o curence ignoring. 2. Does not take into account grammatic correctness of the sentence.