Celine Hajjar
celinehajjar98@gmail.com

Lab session # 2
ALTEGRAD 2021

11/22/21

1 Question 1

According to the presentation, the greedy decoding strategy is very efficient computation- and memory-wise. However, it is heavily sub-optimal, because the decision we make at every timestep cannot be changed.
As an alternative, beam search can give better results because we test K hypotheses at a time and maintain the best combination. It is however computationally heavier.

2 Question 2

One problem that I found with the model is that it doesn't perform well on long sequences. For example:

My brother broke the vase that I bought for my mother on her birthday

↓

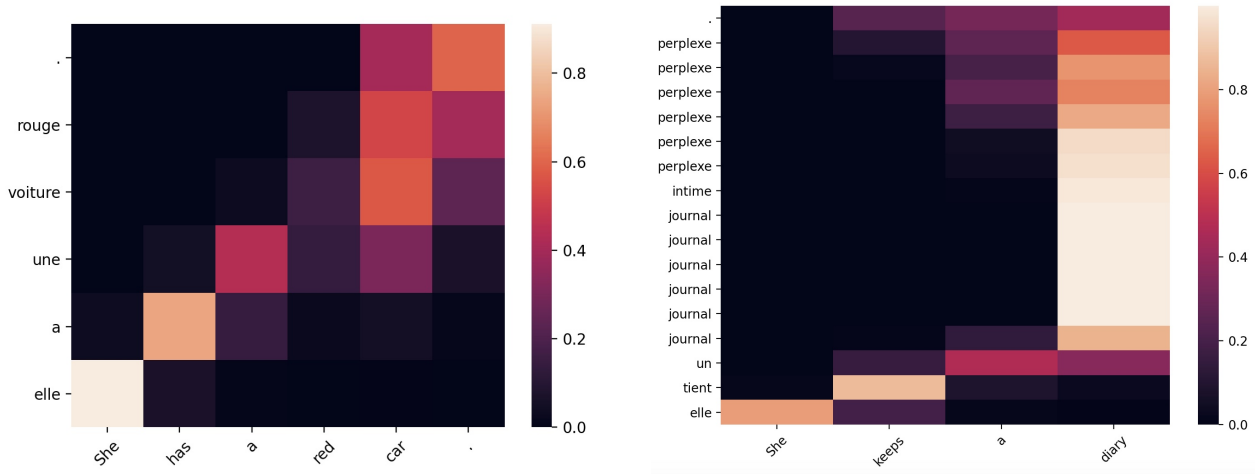
Mon frère a cassé brisé le vase que j j ma ma ma à anniversaire anniversaire .

Also, with the example: **She is so mean** → *Elle est tellement méchant*, we can see that the model fails to infer the gender. Some words are also translated several times (as seen in the test sentences provided in the lab notebook).

There are several ways we can address these issues:

1. As stated in [2], one drawback of global attention is that attending to all words on the source side for each target word makes it impractical to translate longer sequences. Thus, we can try using local attention instead, which consists in focusing only on a small subset of the source inputs per target word.
2. In [4], the authors propose a method to address the repeated translations problems. They use a "coverage vector" that keeps track of the attention history, and allows the model to focus its future attention on untranslated words.
3. In our model, the decoder uses its prediction at timestep $t-1$ as input for timestep t. Alternatively, we could use the "teacher forcing" technique, which consists in feeding the decoder the ground truth from the previous timestep as input, for a fraction ϵ of tokens.

3 Question 3



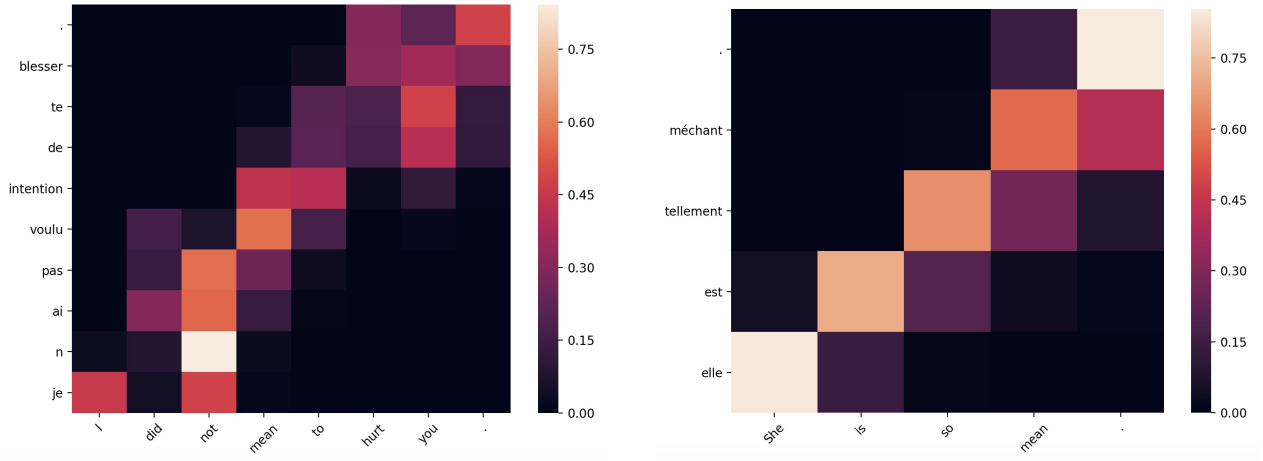
(a) Inversion is captured

(b) Repeated translation

Figure 1: Examples of attention scores for different translations

In order to plot the attention weights, I had to slightly change the implementation of the attention module (to return the scores in addition to the context vector). The `predict` function of the model returns the translated sentence as well as the scores corresponding to the attention of every target word w.r.t all the words in the input sentence.
Two examples of the visualization is shown in figure 1. We can see that the model managed to invert the noun and the adjective in 1a (**red car** became *voiture rouge*). However, the translation is less satisfactory in the second example, where we encounter the problem of repeated translations brought up in Question 2.

4 Question 4



(a) ”I did not mean to hurt you”

(b) ”She is so mean”

Figure 2: The model can capture the context of the sentence.

The score matrices for both examples are shown in figure 2.
A good thing about the model is its ability to understand that the word ”mean” is used in two different contexts, both as a verb and as an adjective. The word ”intention” has high attention weights for both ”mean” and ”to”, which is logical. The overall translation is quite satisfactory, but not perfect: ”voulu intention” is redundant, and ”méchant” is not well *accordé*.
One way to address these issues is to use bidirectional language models, as the authors propose in [3] and later on in the famous paper introducing BERT ([1]). Such models can take into account the context in both directions, and thus learn deep word representations that capture polysemy and complex semantics.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[2] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. *CoRR*, abs/1508.04025, 2015.

[3] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *CoRR*, abs/1802.05365, 2018.

[4] Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. Coverage-based neural machine translation. *CoRR*, abs/1601.04811, 2016.