Martin BRICENO
jbriceno@telecom-paris.fr

Lab session # LAB2
ALTEGRAD 2020

11/30/20

1 Question 1

According to the ACL tutorial presentation, the **greedy decoding strategy**, although it is less computationally expensive, is not the most optimal one we can use since the predictions obtained in each step ”t”, are made maximizing the probability of obtaining this given the preceding words and the context c_t. This is problematic since if the previous words were not well chosen we cannot go back and choose better ones. Because of this the next word we find will not have a chance to give us a good translation in general, even if it maximizes the probability at each step. In other words, maximizing the probability of each word in each step does not ensure the best sentence in terms of coherence and grammar.

A better strategy will be to use the **beam search decoding strategy**, also shown in the presentation, which is more computationally expensive, but it solves the above problem by giving us more optimal results since it does not give us the best option (greedy) at each step but provides us with a set (Beam Width parameter) of better options. In this way, it performs a kind of deep search, testing combinations among the best results to find the best combination over the whole sentence, which allows it to be more flexible and optimal than the previous strategy.

2 Question 2

Let’s see some of the translations obtained with the pretrained model:

- I am a student. ⟹ je suis étudiant . . . .
- I did not mean to hurt you ⟹ je n ai pas voulu intention de blesser blesser blesser blesser blesser . blesser . blesser . . . .
- Help me pick out a tie to go with this suit! ⟹ help me to pick out a cravate to go with this ! ! ! ! ! ! ! ! ! ! ! ! ! ! $< EOS >$
- I can’t help but smoking weed ⟹ je ne peux pas empêcher de de fumer fumer fumer fumer fumer fumer fumer fumer fumer fumer urgence urgence urgence urgence urgence urgence . urgence urgence . urgence urgence .
- The kids were playing hide and seek ⟹ les enfants jouent cache cache cache cache caché caché caché caché caché caché caché caché caché caché caché caché caché caché caché caché caché dentifrice perdre caché risques rapide caché risques éveillés

We can see that some words or symbols are constantly repeated. Although here we show only a few examples, this problem occurs in all translations of our model. We note that this problem is especially evident at the end of the sentence. For those original phrases that ended in ”.”, their translation repeats that symbol until completing the maximum size of the translation (30 words) while for those that do not end with this symbol, we see that what is repeated is usually the last word translated as **”blesser”**. Sometimes, after this last translated word, others appear that are not related to the original phrase, such as **”urgence”**, **”dentifrice”**, **”perdre”**, **”risque”**, etc.

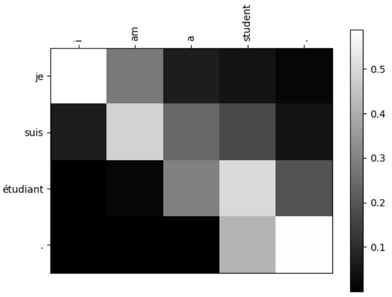
According to the articles [2] and [3], we see that even if we use our attention to provide context to each word we are going to predict at a given step, this is not enough if we do not indicate at the same time which source words have already been translated to take them less into account. In Article [2] we see that one of the consequences of not keeping this record of words translated is **over-translation**, which often involves translating many time the same word, which is our present problem. Therefore, there are two possible solutions to this problem.

In Article [3] they propose the use of a **coverage vector** that indicates at each step which words have already been translated and which have not, in order to pay more attention to them, since the prediction of each new word depends on certain relevant parts of the source phrase. The way to implement this is to create a C vector initialized to 0 and size the number of words in the source phrase. This vector will then be updated at each step to allow to indicate which words have already been translated. We must take into account that this vector must be delivered as input to the attention model so that it adjusts the attention on the words not yet translated, this is called **Input-feeding Approach** by the article [2].

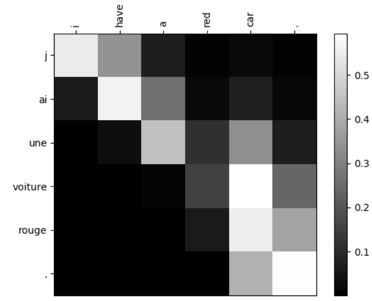
In article [2], the authors propose the use of two types of attention models, the global and the local, we have implemented the global. However, given the above, we need to focus the attention of the model not on all the source words, but on a relevant subset of them in a given step. Therefore, it would be best to use a **local attention model** that takes into account those words that have not yet been translated and with these create the necessary context to make the next prediction.

3 Question 3

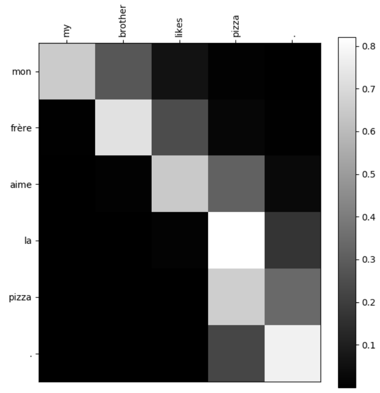
Let's look at the following source/target alignments:



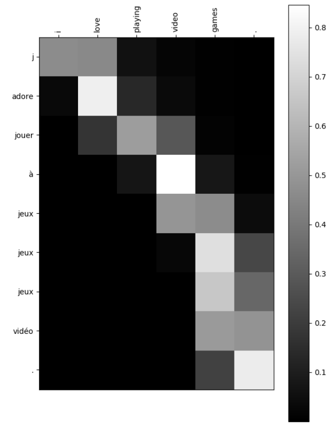
(a) I am a student.



(b) I have a red car.



(c) My brother likes pizza.



(d) I love playing video games.

These graphs were obtained by modifying the "forward" method of the "seq2seqAtt" class to return the weight of the annotation (norm_scores). According to article [1], this variable is essential to display these graphs because it indicates the contribution of each source word to the prediction of a new word.

In figure a, we see for example that the word "he" depends mostly on "i" and "am". From this we get the idea that to predict "je", the contribution of "i" "am" was indispensable.

In figure b we see that the word "une" depends on "a" and "car" since these indicate that it is a preposition and in the feminine gender respectively. We also notice that the word "rouge" depends mostly on "car", without taking into account the word "red". Here we notice that the attention was not so optimal since, even if It made the inversion of words, it was not because the word "rouge" depended on "red".

In figure c, we notice that the words "la" and "pizza" in the translation depend mostly on the source word "pizza", this shows that our translator learned that when we translate to French, some atomic words become binary as in this case.

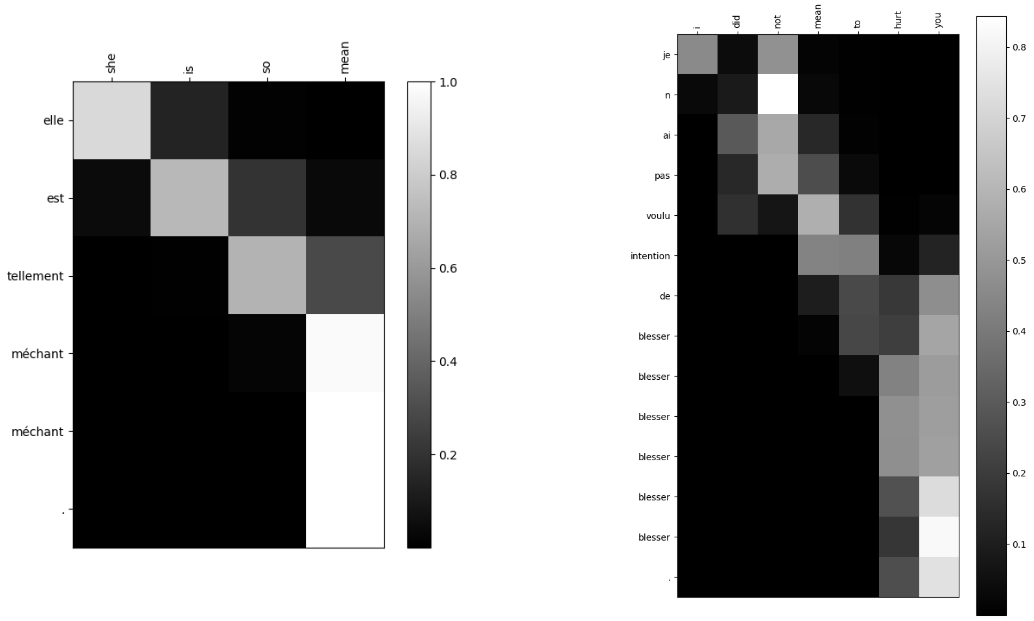
In figure d, we see again that the adjective-noun inversion was not given for the right reason, the word ”vidéo” in French does not depend on the word ”video” in English, but mostly on the word ”games”.

4 Question 4

Let’s look at the following translations:

- did not mean to hurt you $\implies$ je n ai pas voulu intention de blesser blesser blesser blesser blesser blesser . blesser . blesser . . . .
- She is so mean $\implies$ elle est tellement méchant méchant . $< EOS >$

Let’s now look at the following source/target alignments of these sentences.



(a) She is so mean

(b) I did not mean to hurt you

In both sentences we notice that the word ”mean” appears. However, **its translation is different in each sentence**. In the second sentence it is translated as ”mechant” since the translation was only based on itself as we can see in the graph (a) and therefore considered it as an adjective. However, in the first sentence we see that it is translated as a verbal phrase composed of ”voulu” and ”intention” as we can see in the graph (b).

We can say that our **model of attention learned to give context to the words** and therefore we see that the word ”mean” managed to be translated according to the appropriate context, that is why the two different translations.

References

[1] Kyunghyun Cho Dzmitry Bahdanau and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *arXiv preprint arXiv:1409.0473*, 2014.

[2] Hieu Pham Minh-Thang Luong and Christopher D Manning. Effective approaches to attention- based neural machine translation. In *arXiv preprint arXiv:1508.04025*, 2015.

[3] Yang Liu Xiaohua Liu Zhaopeng Tu, Zhengdong Lu and Hang Li. Modeling coverage for neural machine translation. In *arXiv preprint arXiv:1601.04811*, 2016.