BARILLER Halvard
halvard.bariller@ens-paris-saclay.fr

Lab session # 1
ALTEGRAD 2023

10/17/23

1 Question 1

What do you think about our greedy decoding strategy? Base your answer on slides 87-95 from this presentation (taken from this ACL tutorial).

In our context involving a Seq2Seq model with attention, the model produces a probability distribution over the potential next tokens at each step of the generation process. We call "greedy search" the decoding strategy consisting in picking the token with the highest probability at each step and appending it to the output sequence. This method has the non-negligible advantages of computational speed (unlike an untractable exhaustive search) and simplicity. Its computational efficiency as compared to beam search or ancestral sampling makes it a really interesting contender. When comparing the Negative Log-Likelihood (NLL) loss, the greedy search has been shown to be more efficient than these other techniques (due to the fact that greedy decoding takes the output with the highest probability) [2].
However, this approach can lead to suboptimal translations. For example, the phrase "*I am looking forward to meeting you*" might be translated inaccurately to "*Je suis regarder avant de te rencontrer*" because the model might opt for the most probable word ("*regarder*" which means "*look*") at one step instead of the more contextually appropriate phrasing: it thus suffers from a lack of contextual intelligence. Another issue is the lack of diversity: greedy search might repeatedly generate common phrases, making translations sound overly generic. Lastly, an early mistake in decoding can also have a domino effect: in a training not relying on teacher forcing (meaning that each token output by the decoder is used as input for the next, even if this token has been wrongly predicted), if an incorrect word choice is made early on, the rest of the sentence can go awry. For instance, if the model mistranslates "apple" as the fruit "pomme" when it should have been the tech company "Apple", all subsequent tokens might reflect this context error.
Therefore, despite its efficiency, the lack of exploration in greedy search (as compared to the other two) can make it susceptible to the aforementioned pitfalls, especially when nuances and context are crucial, as is often the case in neural machine translation. These weaknesses have been highlighted when comparing the BLEU scores with other techniques [2].

2 Question 2

What major problem do you observe with our translations? How could we remediate this issue? You may find inspiration from reading [3, 5].

The most noticeable issue with the obtained translations is that the model seem to have trouble in predicting the <EOS> token. Indeed, out of the 13 examples given to test the model, only two of them finished with an end-of-sentence token without reaching the imposed limit size of 30 tokens. Instead, the model outputs a succession of the same token (usually a dot, but sometimes repeated words occur as well) without understanding that the sentence should end.
There exists several techniques to mitigate this issue. First, we could introduce a *Coverage vector* [5]. The idea behind using a Coverage vector in attention mechanisms is to address the problem of over-translating and focusing more importantly to certain source positions while neglecting others. The Coverage vector indicates whether each source word is translated or not, thus encouraging the model to pay less attention to translated words and more attention to untranslated words throughout the decoding process.
A second method is the *Input-feeding* approach [3]. This approach has been introduced to reconcile Global and Local Attention models while performing neural machine translation. Indeed, the authors needed a way to mimic a coverage set and keep track of the previous alignment choices while using these two-scaled attention mechanisms.

3 Question 3

Write some code to visualize source/target alignments in the style of Fig. 3 in [1] or Fig. 7 in [3] . Provide and interpret your figures for some relevant examples (e.g. to illustrate adjective-noun inversion).

Here are some examples of alignments showing interesting knowledge the model has gained.

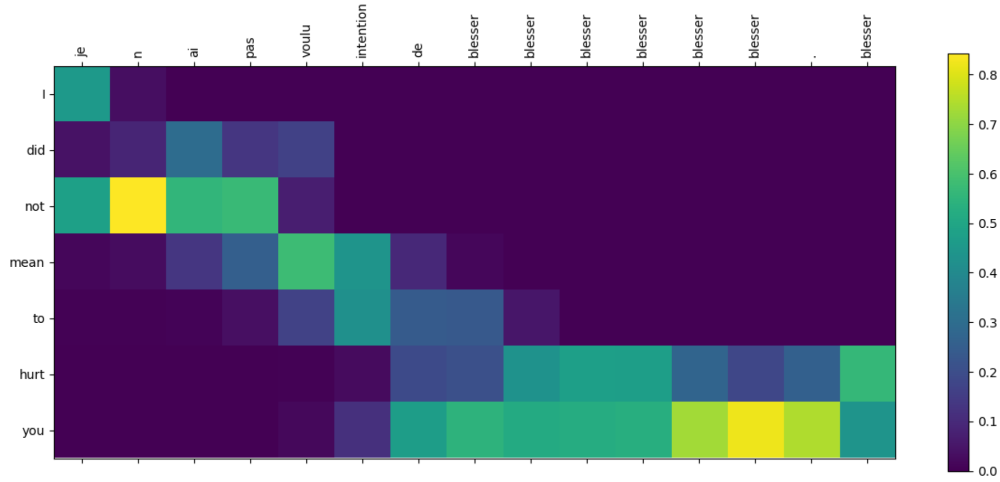

Figure 1: Negation and Inversion

In Figure 1, we see that the model managed to put his focus on the very important word ”not”, without which the sentence would completely change its meaning, and invert the order of the words to take that into account. We see the same principle of inversion in Figure 2. However, we notice that most of the focus when outputting ”rouge” was on car and little on ”red”.

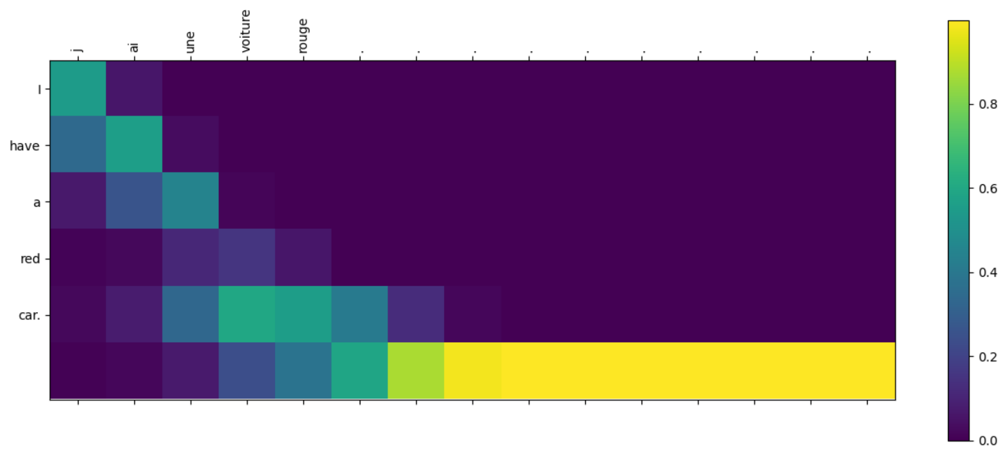

Figure 2: Inversion

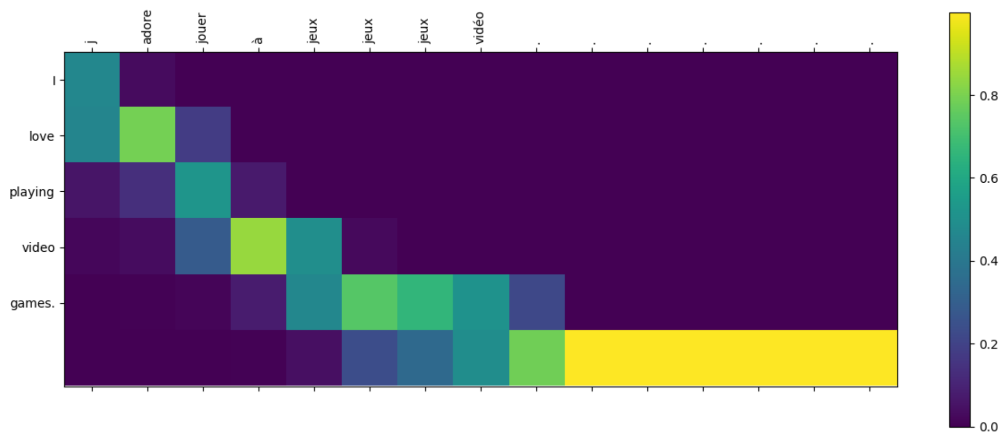

Figure 3: Output Sequence Length

In Figure 3, we notice once again that the model does not seem to have difficulty in predicting a sequence of different length than the input one.

Finally, we notice in Figure 4 that the model managed to accord the adjective in French, whereas it does not exist in English.

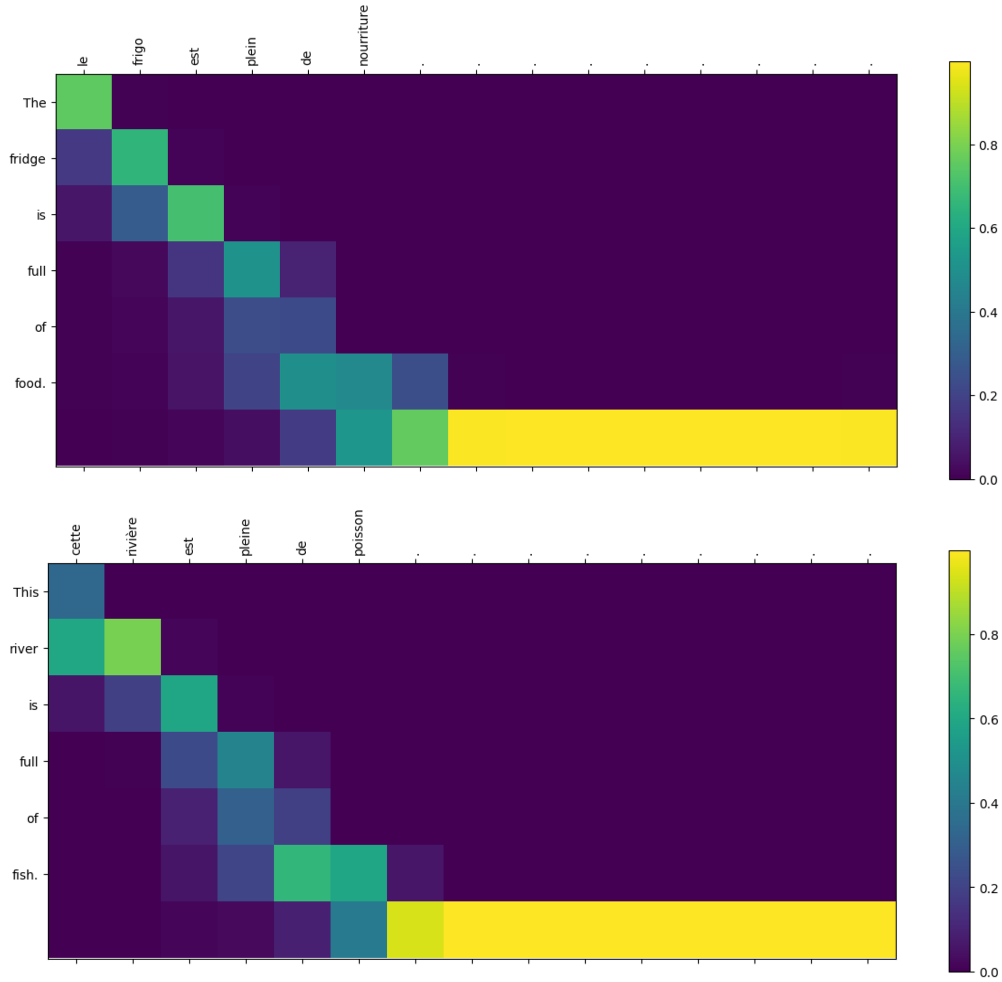

Figure 4: According Adjectives

4 Question 4

What do you observe in the translations of the sentences below? What properties of language models does that illustrate? Read [4? *] to get some ideas.*

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2014. cite arxiv:1409.0473Comment: Accepted at ICLR 2015 as oral presentation.

[2] Junyoung Chung, Kyunghyun Cho, and Yoshua Bengio. A character-level decoder without explicit segmentation for neural machine translation, 2016.

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

[4] 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.

[5] Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. Modeling coverage for neural machine translation, 2016. cite arxiv:1601.04811Comment: Add subjective evaluation on top of ACL version: 25