MERRHEIM Maïssane
maissane.merrheim@telecom-sudparis.eu

Lab session # 1
ALTEGRAD 2023

10/17/23

1 Question 1

The Greedy Decoding method is fast, and efficient, but not always leads to the best translation. At each time step, we generate a token of the highest probability. However, this method sometimes have issues. In fact, choosing the the best token at the current step is not always the best choice because it doesn't always lead to the best sequence.

The beam search is another technique that might be better to generate the best sequence. It works as the following : at each time step, it generates different sequences hypotheses. Beam search allows several hypotheses to be pursued simultaneously and then continues only with the top-K hypotheses. When we arrive at the end of the input sequence, the algorithm chooses the best sequence (highest probability of the sequence). This method allows to consider the best generated sequences and is more precise.
Figure 1 provides a visual illustration of the beam search decoding process. The red arrows and red words correspond to the sequence with the highest probability. In blue, other sequences are generated simultaneously.

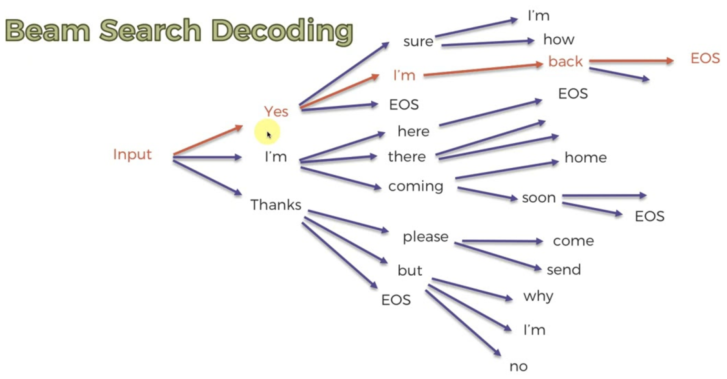

Figure 1: Visual illustration of the Beam Search Decoding [2]

2 Question 2

The major problem we face with this translation is the repetition of the words (most of the time the last words), or the repetition of the syntax ("!" or "." at the end of the sentence).

```
max source index 5281
source vocab size 5278
max target index 7459
target vocab size 7456
= = = = =
 I am a student. -> je suis étudiant . . . . . . . . . . . . . . . . . . . . . . . . . . .
= = = = =
 I have a red car. -> j ai une voiture rouge . . . . . . . . . . . . . . . . . . . . . . . . .
= = = = =
 I love playing video games. -> j adore jouer à jeux jeux jeux vidéo . . . . . . . . . . . . . . . . . . . . . .
= = = = =
 This river is full of fish. -> cette rivière est pleine de poisson . . . . . . . . . . . . . . . . . . . . . . . .
= = = = =
 The fridge is full of food. -> le frigo est plein de nourriture . . . . . . . . . . . . . . . . . . . . . . . .
= = = = =
 The cat fell asleep on the mat. -> le chat s est endormi sur le tapis . . . . . . . . . . . . . . . . . . . . . .
= = = = =
 my brother likes pizza. -> mon frère aime la pizza . . . . . . . . . . . . . . . . . . . . . . . . .
= = = = =
 I did not mean to hurt you -> je n ai pas voulu intention de blesser blesser blesser blesser blesser blesser . blesser . blesser . . . . . . . . . . . . .
= = = = =
 She is so mean -> elle est tellement méchant méchant . <EOS>
= = = = =
 Help me pick out a tie to go with this suit! -> aidez moi à chercher une cravate pour aller avec ceci ! ! ! ! ! ! ! ! ! ! ! ! ! ! <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
= = = = =
 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 ca
= = = = =
 The cat fell asleep in front of the fireplace -> le chat s est en du du pression peigne peigne cheminée portail portail portail portail portail portail portail po
```

Figure 2: Translated sentences with greedy decoding

Zhaopeng Tu et al, 2016 in their paper ”Modeling coverage for neural machine translation” [**?**] also notice this problem with the NMT model. The NMT model tend to over-translate (repetition of the translated words) or under-translate words (some words aren’t translated), because it doesn’t maintain a record of the words that have already been translated. To face this issue, they add a ”coverage” vector. This vector has this aim to keep track of which words have already been translated, so that the model don’t over-translate the words.

Let’s say we have an input sentence of five words x = $\{x_1, x_2, x_3, x_4\, x_5\}$. We construct a vector C that corresponds to the ”coverage” vector and that keeps track of the translated words. At the beginning of the algorithm, no words have been translated so C= {0,0,0,0,0}. When the first two words are translated for example, C= {1,1,0,0,0}. Thanks to this vector, the model can then know which word it has already translated and won’t translate it again. The source sentence is fully translated when the coverage vector is equal to C={1,1,1,1,1}.

This coverage vector is added to the source sentence’s information (initialized to the null vector) and will help the attention model to put more focus on the untranslated source words, so that is doesn’t over translate. According to the paper, this method has enhanced the NMT model and provided better translation.

3 Question 3

Here are some examples that illustrates the inversion between a noun and an adjective :
I decided to translate this simple sentence : ”She has short hair”. The translation given by the model is the following :
”elle a de cheveux courts . . . .”
Let’s take a look at the English sentence. The adjective that is related to the hair is ”short”, and comes before the noun. However, if we translate this sentence in French, the adjective comes after the noun : ”Elle a des cheveux courts”. The NMT model understands that it needs to do this inversion by giving attention to the words ”short” and ”hair”. Therefore, in the alignment matrix, we notice that the scores between the word ”cheveux” and the word ”short” is high, and the scores between the word ”courts” and ”hair” is also high.

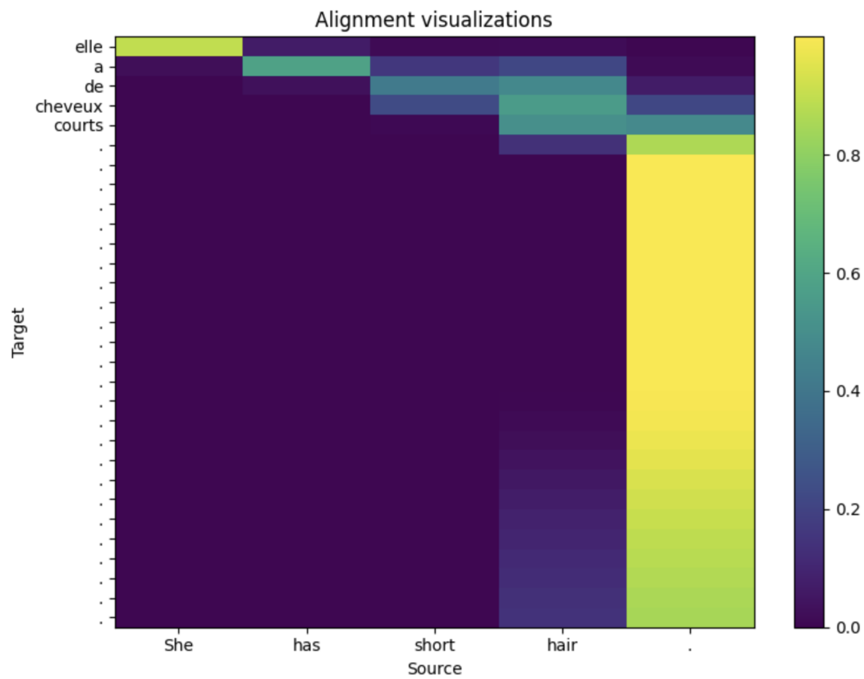

Figure 3: Alignment visualisation for the sentence ”She has short hair”

We also observe the same phenomenon with the sentence given by the assignment ”J ai une voiture rouge”. In this example, we also notice that the scores of attention between ”voiture” and ”red” and between ”rouge” and ”car” are high, because the model knows that these two words are strongly related in the syntax of the sentence.

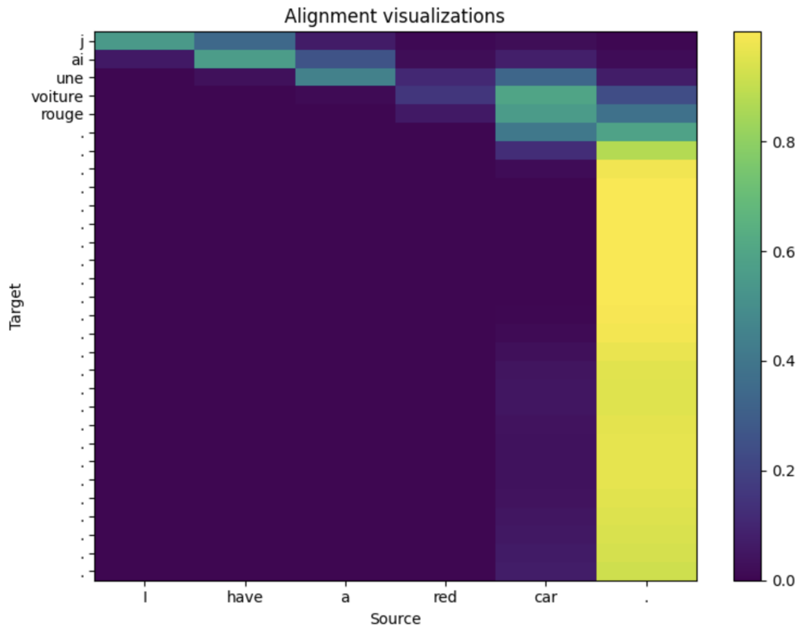

Figure 4: Alignment visualization for the sentence "J ai une voiture rouge"

The visualisation of the translation of "my brother likes pizza." is also interesting. Thanks to the alignment matrix, we can notice that the attention given to the word "pizza" in the English sentence, is the same for the determinant "la" and the translated word "pizza" in the french sentence.

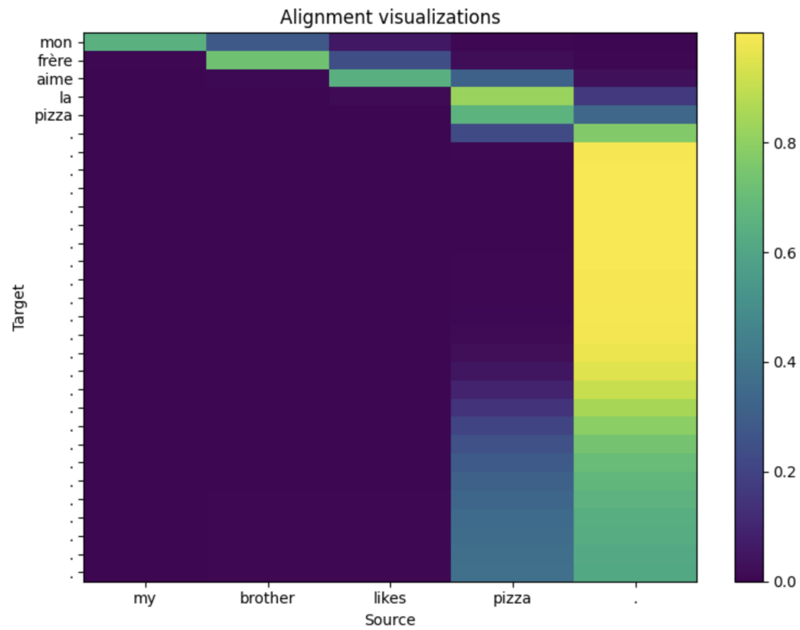

Figure 5: Alignment visualization for the sentence "my brother likes pizza"

4 Question 4

The two sentences "I didn't not mean to hurt you" and "She is so mean" have a word in common : the word "mean". However, according to the contextualization of the word, the translation of it is completely different? in the first sentence, the word "mean" refers to an intention, whether in the second sentence, the word "mean" means naughty, or bad. Therefore, the contextualisation of the words are very important, because it can radically change the meaning of the sentence. What's more, we can also highlight the fact that for the translation of the sentence "She is too mean", the model didn't take into account that the subject of the sentence is a female and didn't do the agreement between the noun and the adjective in french. It translated

”mean” by ”méchant” instead of ”méchante”.

These two phenomenons (the meaning of a word depending on its contextualisation, and the polysemy of words) are what the papers ”Deep contextualized word embeddings” written by Matthew E. Peters [1] and Al. and ”Bert: Pre-training of deep bidirectional transformers for language understanding” by Jacob Devlin and Al. [3] points out. In fact, the contextualisation of the word embeddings is necessary to provide a correct translation that takes into account the polysemy of the words (e.g., “méchant”, “méchante” as adjectives, “méchanceté”, as noun, and “méchamment ” as adverb), the syntax and the semantic. Therefore, in these papers, the authors provide contextual word embeddings to enhance the translation and to take into account the context, the polysemy and the semantic of the words in the sentence.

References

[1] Kenton Lee Jacob Devlin, Ming-Wei Chang and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *arXiv preprint arXiv:1810.04805*, 2018.

[2] Minh Nguyn. Visual illustration of the beam search decoding. In *11 Beam Search Decoding, https://i.ytimg.com/vi/BvQHggTHaLQ/maxresdefault.jpg*, 2021.

[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.