# Data Augmentation using Clause Segmentation and Synthetic Translation

Taycir Yahmed

July 2018

Most public parallel corpora are formed of long sentences. Consequently, neural translation models tend to generate a long output with n-grams repetition, even when they are exposed to a short sequence or a one-word example. This causes the repetition problem, explained by the fact that none of the neurons learns the representation of length, thus the model generates a long sequence by default. In other terms, the probability of appearance of the end-of-sentence token <eos> will not be high enough to stop the output generation when translating a short sequences.

| **Source** | et il croît depuis lors à un taux de 5 % |
|---|---|
| **Baseline translation** | since then and since then at 5 % at 5 % |

Table 1: Illustration of n-grams repetition on clauses translation

To solve this problem, a possible solution is augmenting the training parallel corpus with sequences of a smaller length, typically one-word examples (using bilingual dictionaries) and sub-sentences. To generate the sub-sentences, two important steps are considered:

- First, detect and segment clauses in long sentences.
- Second, retrieve the clauses exact translation.

## Clause detection and segmentation

In neural machine translation, sentences with more that 50 tokens are usually dropped. According to many research papers, sentences with such length harm the performance [1]. As a consequence, an approach is suggested to segment these sentences to clauses and thus use them while training instead of simply dropping them.

The first task is detecting clauses in long sentences. To do so, linguistic rules, specific to each language that mark the beginning / end of a clause, are needed. These rules are formulated within a *Treebank*.

In linguistics, a treebank is a syntactic or semantic sentence structure annotator. The introduction of the first parsed corpora in the 90s, revolutionized computational linguistics, particularly after publishing *Penn Treebank* [2], the first large-scale treebank. Indeed, annotated treebank data has been crucial in syntactic research to test linguistic theories of sentence structure. In addition, there are variants of treebanks, including phrase structure annotators and dependency structure annotators. Note that in these experiments, phrase structure annotators are used.

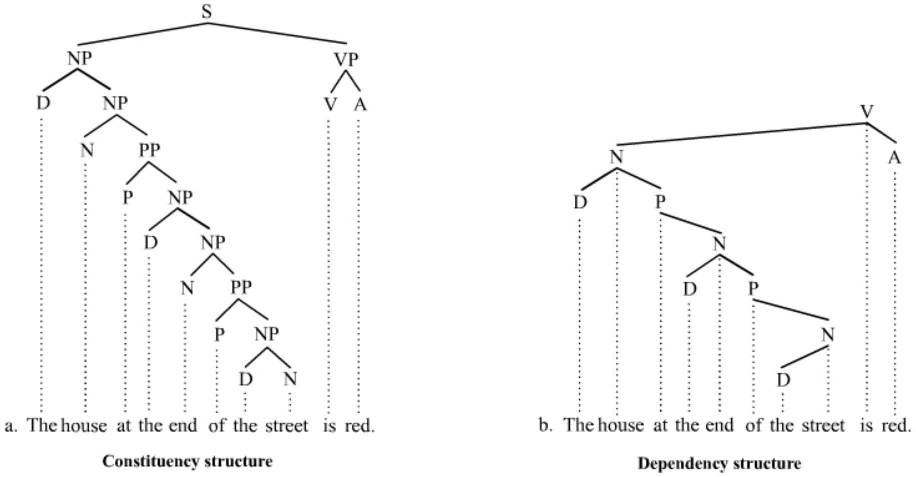
S
NP
VP
D
NP
V
A
N
PP
P
NP
D
NP
N
PP
P
NP
D
N
a. The house at the end of the street is red.
Constituency structure
V
N
A
D
P
N
D
P
N
D
b. The house at the end of the street is red.
Dependency structure

Figure 1: Variants of syntactic treebanks

Since this experiment deals with French to English translation scenario, a French Treebank is needed. Due to license constraints and the need for phrase annotators, Paris 7 French Treebank was chosen. This Treebank was initiated in 1997, with the collaboration of IUF, CNRS and CNRTL. It consists of 1 million words of the newspaper Le Monde (1989-1995). The full list of the generated tags is accessible here [3].

**Clauses segmentation:** The first step is identifying the usually dropped sentences, those with more than 50 tokens (words). Afterwards, these sentences are annotated each with phrase tags using the French treebank.

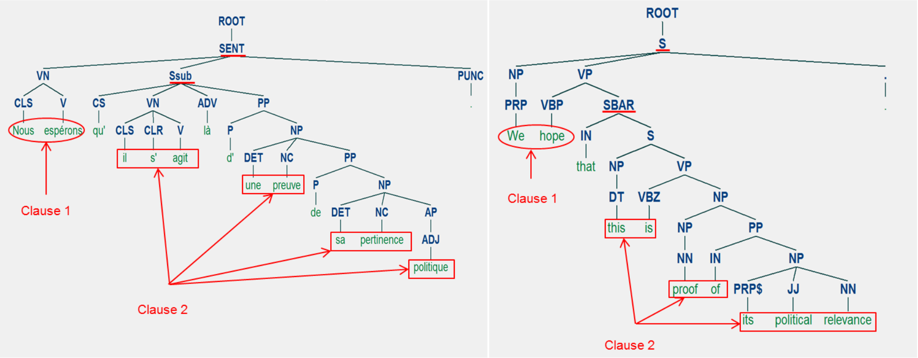

Figure 2: Clause detection and segmentation: French and English examples

To select the clauses, specific tags are selected:

- Selected tags for English:
  - S: simple declarative clause, i.e. one that is not introduced by a subordinating conjunction or a wh-word and that does not exhibit subject-verb inversion.
  - SBAR: Clause introduced by a subordinating conjunction.
- Selected tags for French:
  - Ssub: subordinate clause (“complétive”, indirect interrogative, circumstantial subordinate)
  - Sint: clause “conjuguée interne” (coordinated, direct speech, incise)
  - PP: prepositional phrase
  - Srel: relative proposition (starting with a relative pronoun)
  - COORD: coordinated phrase
  - VPinf: infinitive proposition (starting with a preposition)

Using these tags, the long sentences are segmented to the clauses that form them. So, whenever a new tag is encountered, when visiting the different nodes of the parsing tree, a new clause is generated.
This segmentation step results in a corpus of short sequences in the source language. Now, the exact translations for these clauses have to be generated.

## Synthetic translation of the extracted clauses

To translate the clauses, the original model can’t be used because it doesn’t handle short sequence translations and would generate n-gram repetition. However, in this section, a method allowing quality translation for sub-sentences is presented. Eventually, this proposed approach generates a bilingual corpus of short sequences / phrases.

Below are the different steps applied to get the clauses’ translation:

1. Using the original model, translate the original long sentences, from which we previously extracted the clauses.

2. Extract the attention weights generated by the previous translations.

   The attention weights are denoted $\alpha_{ij}$, representing the contribution of word i on the source side in the translation of word $j$ in the target side. Note that $i$ ranges from 1 to length of the source sentence, here denoted n; $j$ ranges from 1 to length of the target sentence, here denoted m.

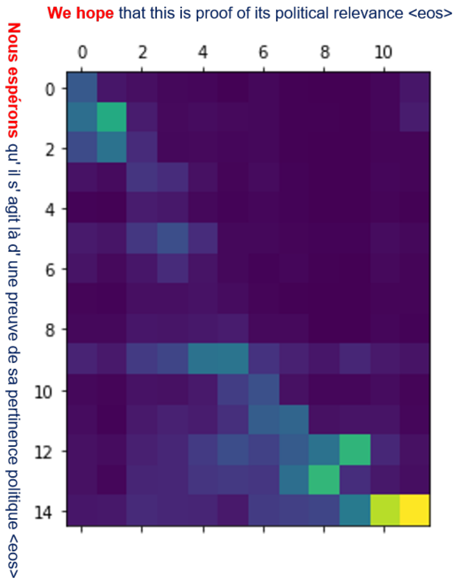

Figure 3: Attention weights generated with translation

Here the source sentence is *"Nous espérons qu' il s'agit là d'une preuve de sa pertinence politique."* and the target prediction is *"We hope that this is proof of its political relevance.".*

Each cell $\alpha_{ij}$, where $1 \leq i \leq n, 1 \leq j \leq m$: n being the length of the source sentence and m being the length of the target sentence, represents the contribution of target word j in the translation of the source word i. Note that the lighter the cell, the more important the attention weight.

**Important remark:** Here, the matrix is predominantly **diagonal**: this indicates how much the French and English languages are aligned. An example of the attention matrix corresponding to non-aligned languages can be seen in the below figure. Furthermore, some **anti-diagonal** sections in the matrix can be observed, these are due to the difference in the order of adjective compounds between French and English, e.g. *"pertinence politique"* is translated to *"political relevance"*.

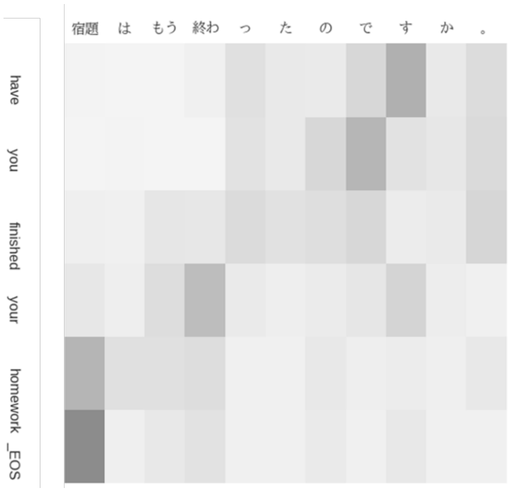

Figure 4: Attention matrix of Japanese to English translation

3. Apply the following algorithm to retrieve the clauses' translation.

```
Result: Translate clauses synthetically
Input: source clauses generated in previous section.
for clause in source clauses do
    n ⟵ length of ss, the corresponding long sentence from which the
      clause is extracted.
    m ⟵ length of ts, the translation of ss.
    $\alpha_{ij}$ ⟵ attention weights, with $1 \leq i \leq n, 1 \leq j \leq m$.
    ind ⟵ the clause's position in ss.
    $b_{ij} \longleftarrow \alpha_{ij} * \mathbb{1}_{i \in [ind, ind+c]}$ where c is the length of the clause.
    sp ⟵ $\sum_{i=1}^{c} b_{ij}$
    start ⟵ min [ index ( 0.4 * max(sp) ) ]
    end ⟵ max [ index ( 0.7 * max(sp) ) ]
    clause's translation ⟵ ts [ start : end ]
end
```

**Algorithm 1:** Clauses synthetic translation

**Note:** the thresholds 0.4 and 0.7 are selected after experiments on the alignment between French and English languages.

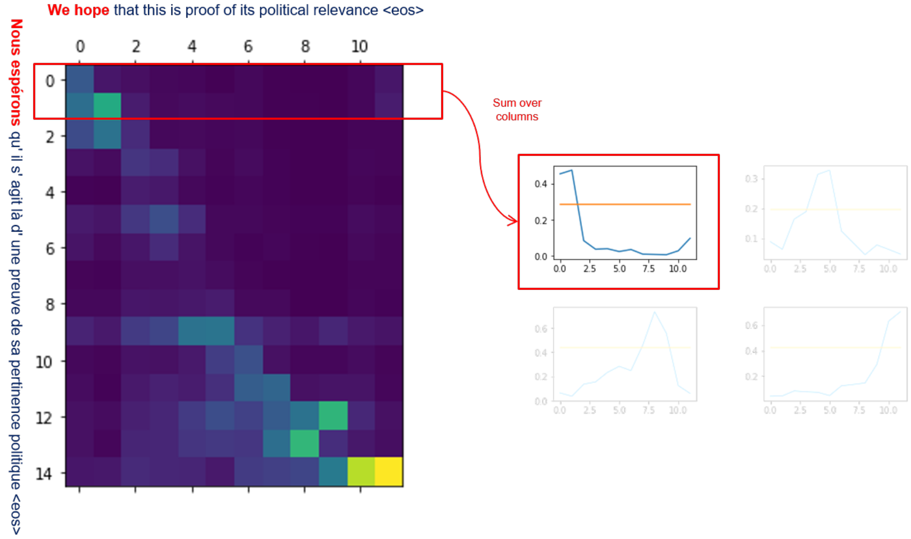

Figure 5: Illustration of synthetic translation of the first clause *"Nous espérons"*

Note that, in the graph on the right, the horizontal axis represents the position of words in the target sentence and the vertical axis represents the contribution of the corresponding target word in the translation of the clause (here: *"Nous espérons"*).

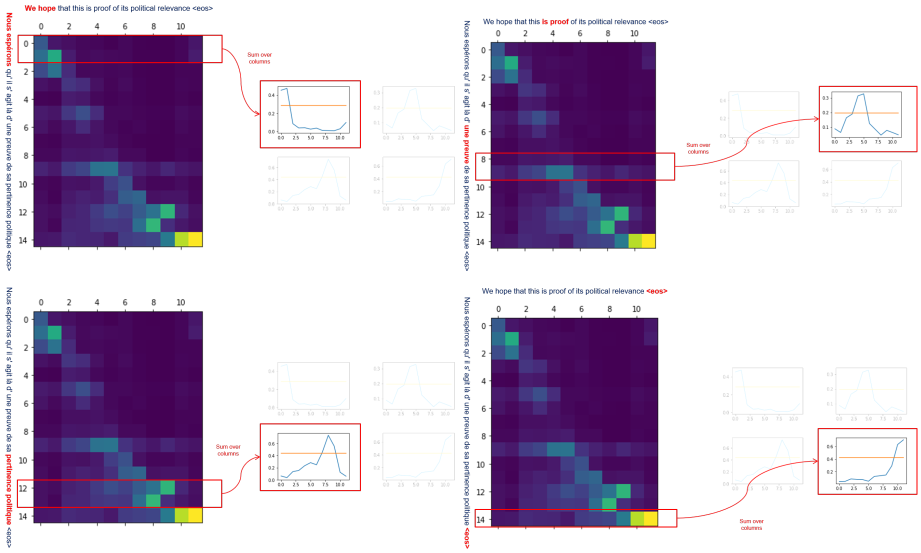

Figure 6: Information transfer through the source and target sentences.

### Model training with clauses

Using the previously described processes, a bilingual corpus of clauses is constructed. However, in the following experiment, only 35,821 clauses are used, which makes around 3% of the available clauses. Furthermore, each set of clauses is concatenated to the corresponding corpus among the source (French) and the target (English). Afterwards, the two corpora are jointly randomized so that the clauses are not located just in the end of the data set, but spread along the corpus. Then, the model is retrained during 13 epochs with the baseline setup: 2.5 million parallel sentences, 4 bidirectional LSTM attentional encoder-decoder architecture with 500 as embedding size, 500 as number of hidden units and 5 as beam size.

### Results and discussion

Below, I state the scores obtained using this method on WMT 2015 test set and on a test set of clauses out-of-sample.

| **Experiment** | **WMT BLEU** | **Clauses BLEU** |
|---|---|---|
| Baseline | 28.41 | 49.73 |
| Augmented model with bilingual clauses | 28.85 | 58.31 |

Table 2: Clauses integration scores

**Quantitative discussion:** Integrating the clauses improves the performance with 0.44 BLEU on WMT 2015 and 8.58 BLEU on a test set of clauses. Note that in this experiment, only 3% of the available clauses are used.

**Qualitative discussion:** Integrating the clauses has an influence mostly on translating short sequences. The method was suggested to solve the problem of n-gram repetition and indeed it did. Below is an example illustrating how the augmented model translates short sequences.

| | |
|---|---|
| **Source** | et il croît depuis lors à un taux de 5 % |
| **Baseline translation** | since then and since then at 5 % at 5 % |
| **Augmented model** | and it has been growing since then at a rate of 5 % |

Table 3: Clauses integration qualitative results

## Acknowledgments

Many thanks to Bruce Delattre and Edouard D'Archimbaud for their constructive advice and valuable feedback.

# References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014.

[2] A. Bies. Bracketing guidelines for Treebank II style — Penn Treebank project. Technical report, Department of Linguistics, University of Pennsylvania, Philadelphia, PA, USA, 1995.

[3] Corpus arboré pour le français / french treebank. `http://www.llf.cnrs.fr/Gens/Abeille/French-Treebank-fr.php`. Accessed: 2018-06-14.