Comparison of different methods for Machine Translation from French to English
Machine Learning for Natural Language Processing 2022

Claire Acciari
ENSAE Paris
`claire.acciari@ensae.fr`

Caroline Moreau
ENSAE Paris
`caroline.moreau@ensae.fr`

Abstract

This project explores Machine Translation from French to English. We used the Europarl corpus, a collection of transcriptions of proceedings of the European Parliament in eleven languages. Our baseline consists of two vanilla RNNs with one hidden layer for the encoder and decoder. We then tested adding a Gated Recurrent Unit layer to both, and tried using a Transformer model and a pretrained model: BART. Mostly, the models perform poorly, either from too much simplicity or lack of necessary training.

All the code for this project is available in a Google Colab notebook.

1 Problem Framing and Data

Our project consists in testing out and comparing different models for Machine Translation. We chose to test translation from French to English to be able to check the quality of the translations ourselves.

The data we used is the EuroParl corpus. It consists of the written transcriptions of proceedings of the European Parliament. The transcriptions are available in most languages of the EU.

Data Exploration and Preparation

The original text files comprise more than two million sentences each. Because we don't have access to much processing power, we kept only the first 20,000 lines and treated each one as a sentence.

We tokenized the sentences using the Tweeter Tokenizer. We additionally used a Phraser to detect multi-word expressions and include them in the tokens. Then we could build a vocabulary and plot the number of occurrences of each word to verify Zipf's law. The result is in Figure 1: Zipf's law is verified. Finally, a basic exploration conducted using `pandas-profiling` shows that tokenized sentences have a length distributed between 0 and 100, with most lengths between 10 and 60 for both languages. Also, as expected, the length of tokenized sentence in French is strongly correlated (coefficient is higher than 90%) to the length of its tokenized translation in English.

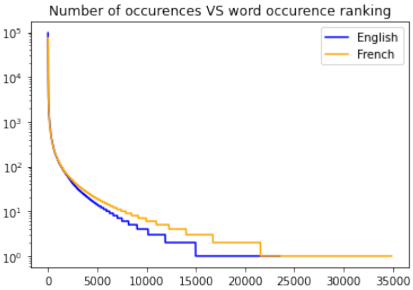

Figure 1: Verification of Zipf's law

2 Experiments Protocol

We used four models. As a baseline, we trained a Vanilla Recurrent Neural Network (RNN) with one layer. To improve a bit on it, we used the model of the TP, where the RNN is replaced with a Gated Recurrent Unit (GRU) with one layer. Then we trained or fine-tuned two Transformer models, which we used in combination with a Word2Vec model for the embedding. The first one is untrained and consists of an encoder-decoder architecture with six layers in the encoder and the decoder. The final model is a combination of an encoder and a pre-trained BART model, which acts as the decoder part of the encoder-decoder.

We trained and validated the models using the same functions as in the TP, or modified versions of those, computing the cross Entropy Loss and the BLEU score to assess the performance of our models. The learning rate was 10^{-4}. The two RNN models and the first Transformer model could be trained on ten epochs. Unfortunately, due

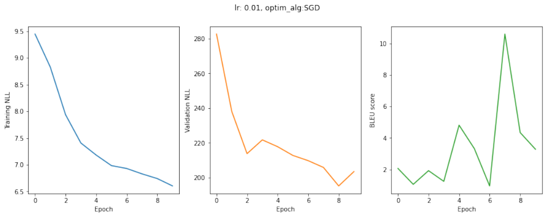

Figure 2: Measures for RNN

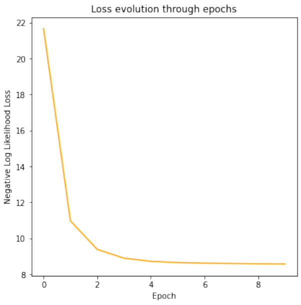

Figure 3: Measures for Transformer

to lack of processing power needed for fine-tuning the BART decoder, we could only run two epochs, and didn't fine-tune the whole model (with the added encoder). We are aware this brings the performance of the model down, but it took too long to run. We also printed the then first target and predicted sentences to evaluate the performances ourselves.

3 Results

Our models globally perform poorly. The measures for the tenth epoch of training are at Table 1, and the same measures for the Vanilla RNN model and Transformer model through training are plotted at Figure 3 and Figure **??**.

	T. loss	V. loss	BLEU
Vanilla RNN:	6.60	203.53	3.28
GRU RNN:	4.38	52.36	16.56
Transformer:	8.57	49.86	13.92
BART:	-470	-361.53	2.31

Table 1: Measures for each model at last epoch or on a test set

The twot RNNs are particularly bad translation models. We see quite an improvement from the Vanilla RNN to the GRU RNN: the validation loss plummets while the BLEU score is multiplied five-fold. However, the GRU RNN is still a bad model: the BLEU score is only of 16.56 out of 100. Although the models would probably benefit from training for more epochs (the BLEU score does not seem very stable for one thing), the training loss is quite low, so the models might tend to overfit if we train them for much longer. This may be linked to the small size of data we had to impose. The fact that both those models are really bad is also revealed when taking a look at some predictions they make: they always predict the word ”the” as a translation for everything. It is the most common word in the English corpus, so this makes sense, but it is completely useless as a way of translation.

The Transformer model behaves poorly as well in terms of loss and score, but when examining the predictions, we notice that it predicts other words than ”the”, which to us is an improvement. The resulting sentences might be complete gibberish but the model learned to associate some other words.

Finally, the model with a BART encoder performs particularly bad, even compared to the much simpler models. But we think it is very probably due to the fact that we could only half train it, and only on two epochs.

4 Discussion and Conclusion

None of our models could be realistically used. This is not really a surprise as Machine Translation usually requires a large amount of data and high model complexity. We defined the first two models as two very simple models to see if there could be an improvement with more advanced models such as the last two. We do see that they is an improvement, although they do not translate very well. The two RNN models could be much improved, simply by adding layers for example. Additionally, the model with the pre-trained BART encoder performs poorly (worse than the untrained Transformer), but we could only run the fine-tuning on two epochs, which is obviously not enough. Training it for much longer would certainly cause a major improvement in the results. Finally, we suspect there might be an imbalance in the dataset, with much more weight in the frequent words.

In conclusion, we verified the almost evident assumption that complex models outperform very simple ones. However, we observe that lack of training (or even fine-tuning, in the case of pre-trained models) and the scarcity of training data cause all those models to behave quite poorly.

References

Jörg Tiedemann. 2012. Parallel Data, Tools and Interfaces in OPUS. *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*

M. Lewis, Y. Liu, N. Goyal et al. 2019. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. arXiv:1910.13461