Unpopular Translation Request: One Word ~~Away~~ Off

Maxime Daigle
DIRO
Université de Montréal
Montréal, QC

Annabelle Martin
DIRO
Université de Montréal
Montréal, QC

Kun Ni
DIRO
Université de Montréal
Montréal, QC

Marc-André Ruel
DIRO
Université de Montréal
Montréal, QC

1 Introduction

In machine translation, some languages pairing are more rich in both sources and varieties of samples. However, for languages where the need of sentence to sentence or word to word translation is a lot less prominent, parallel corpus are usually harder to find and contains less examples. Therefore, it is not always possible to use the state-of-the-art NMT model that requires very large parallel corpus. For these language pairings, we evolve in a low resource environment and we need to improve models based on augmented data or unaligned corpus with unsupervised models.

In this paper, we applied different methodologies on parallel and non-parallel corpus in French and English. Although it is possible to find a large corpus of parallel French-English sentences, the parallel corpus was kept artificially small at 11,000 samples. However, we also make use of unaligned monolingual corpora: an English corpus of 474,000 samples and a French corpus of 474,000 samples. By limiting access to aligned samples while having a large unaligned corpora, we were able to simulate a low-resource NMT environment to train a model to translate from English with no punctuation to French translation with correct punctuation.

The bilingual evaluation understudy (BLEU) score, based on 4-grams, was used to evaluate the results of each model tested. This means that each 4-grams of the translated sample are compared to the 4-grams present in the target sample, and the more 4-grams present, the higher the score is. The BLEU score correlates well with human judgment in translation and is the most common score used to evaluate the performance of NMT model.

2 Related work

Since our corpus of parallel data is relatively small compared to the monolingual corpus, we reviewed methodology that makes use of monolingual corpus such as Artetxe et al. [2017]. They used an encoder-decoder architecture with a bidirectional RNN-GRU as the encoder and a simple RNN-GRU as the decoder. They build their model to be able to train both French to English and English to French translation. The same encoder is used for both French and English as previous work applied (Ha et al. [2016]). They also used pre-trained cross-lingual embedding that were kept fixed during training. Several unsupervised methods can be used to learn those embeddings. Both language would need to be embedded using the same embedding method for both language to reside in the same vector space. Then, the encoder learns to encode both languages in a language-independent fashion and the decoder learns to reproduce the input while at inference level, the decoder is switched to the target translation language. To avoid for the decoder to only learn a word-to-word substitution, the authors add noise to the input by swapping the order of the words in the input. They also make use of on-the-fly translation using the decoder to translate each mini-batch and trained the model on those substituted parallel samples. With their completely unsupervised model, they obtained a BLEU score on English-French translation of 15.13 and with their semi-supervised model, they obtained 17.34 and 21.74 with a parallel corpus of 10,000 samples and 100,000 samples respectively.

We also reviewed the procedures followed by Sen et al. [2019]. They followed a similar procedure as Artetxe et al. [2017] but they made use of multi-tasking by training their model to do translations between 4 different languages. As our simulated context does not allowed for multiple languages, we were not able to make use of it. However, they confirmed again that using a denoising autoencoder and applying back-translation during training does improve the BLEU score of the NMT model. They add the noise within their samples by randomly swapping two adjacent words and then learn the original sentence through the decoder. They also performed cross-lingual embedding with only monolingual corpus. They start by learning two monolingual embedding spaces (X and Y) and use adversarial training to learn translation matrix (W) to learn a map from the first embedding space to the second while training a discriminator to discriminate between WX and Y. The monolingual embeddings are trained using fast-Text with the skip-gram model. Their final model obtained a BLEU score of 13.71 on English to French translation with only monolingual data.

Ren et al. [2019] built upon the ideas from the previous papers. They used a similar model but they also made use of statistical machine translation (SMT) to help with de-noising and help guide the model during the back-translation process. Their SMT is built based on pre-trained language model and the STM and NTM are trained together in a expectation maximization framework. This method is supposed to help the NMT generate better pseudo data and the SMT extract sentences of higher quality. They were able to obtain a BLEU score of 28.92 in English to French translation which is a big improvement compared to the other models described above. However, we did not follow this methodology since it is a very complex implementation compared to the timeframe we had for our experiment.

From these papers, we can see that we can train relatively good NMT model based only monolingual corpus, however, from Artetxe et al. [2017], we know that the models can gain from using some parallel data with a semi-supervised training. In our environment, we have access to only 10,000 samples of parallel data in our training set. Therefore, to improve our model from parallel corpus, we reviewed Fadaee et al. [2017] and Imankulova et al. [2017] which explore data.

Fadaee et al. [2017] focus on data augmentation for low-frequency words which is an issue that is particularly present in low-resource environment. They focused on words that are poorly modeled, mostly words that rarely occur. Their technique consist into finding a sample that contain a common word that can be substitute directly by a rare word from the subset while trying to preserve the plausibility of the new augmented sample. They train an LSTM model forward and backward on monolingual data. For the rare word substitution, they based their substitution on the conditional probability over the vocabulary from both the backward and forward model. If the same rare word appears in the top conditional probabilities for both the forward model and the backward model, the substitution is made. For the selection of the translation, they used automatic word alignment and found the optimal probability to select the right translation. If the word is unaligned or if the highest probability is still below a threshold, the augmented sample is discarded. They went over the parallel sample multiple times and limited the number of time a rare word can be augmented to ensure that a large number of rare words were being included. They were able to improve the BLEU score compared to their baseline by more than 2 points on all their test sets. Unfortunately, their methodology requires multiple models. For our current experiment, we try to create a model that does not require any pre-trained model and the timeline available to complete all our modeling and training doesn't allow to train as many models.

On the other hand, Imankulova et al. [2017] works with filtered pseudo-parallel corpus. They make use of back-translation as Artetxe et al. [2017] did, but they apply filtering to this new corpus and bootstrapping of the translation model. They use a similarity score between the original sample and the back-translated one. They set a threshold and only keep the samples the with highest score. With those, they take the sample created from the first translation (target to source) and use that filtered new source sample as training data. As the model we are using in this paper can do both French to English and English to French translation, we could make use of this technique. With this technique, combined with their bootstrapping methodology, they were able to obtain good BLEU score with both Russian-Japanese and English-Russian translation task.

As one of our task was to ensure that the punctuation of the translated sentence was correct, we also researched models that were used to insert punctuation into text. Cho et al. [2017] used and attention encoder-decoder based model to insert punctuation in a real-time language translation system. Their model is comprised of a bi-directional RNN for the encoder and a RNN model for the decoder. They used the byte-pair encoding to represent their vocabulary. They ensure that the model would not include punctuation in between two of the smaller token created by the byte-pair encoding by using a tag-based representation. They were able to obtain BLEU scores around 13-14 with a word embedding of size 256, a multi-layer bidirectional LSTM with 256 hidden units as an encoder, and a conditional GRU with 512 hidden units as a decoder.

Tilk and Alumäe [2016] used a similar model to insert punctuation. They also used a bi-directional RNN with GRU cells and an attention mechanism. Again, all layers contained 256 hidden units. They were able to obtain F-score above the prior state-of-the-art models. However, they also worked with a smaller vocabulary size, they kept only the words that appeared at least twice which means a vocabulary size of 27,244 words. They also limited the number of possible punctuation by mapping some punctuation to a period, like the semi-colon and colon for example and removed other punctuation altogether.

3 Data

For the purposes of our simulated low-resource machine translation task, we limited our access to data even though there is many source of parallel data between English and French. We have access to a small set (11,000) of parallel samples. However, the English version does not contain any punctuation or capital letters while our French version does. The objective for the translation will also be to go from an English sample with no punctuation to the French translation with punctuation.

Further more, we have access to two monolingual corpus, one in French and one in English. They both contain 474,000 examples. They are not aligned but they both contain the proper punctuation. The two monolingual corpus have not been tokenized. It needs to be done in pre-processing.

We split our aligned dataset into 3: a training set of 9,000 samples, a validation of 1,000 samples and a test set of 1,000 samples. The monolingual data is only used in training and embeddings.

No other data have been used to preserve the low-resource problem environment. This also means that no pre-trained model or embeddings were used. This way, we ensured that our methodology and techniques could apply to languages that have not been worked on yet in papers or by companies specializing in NMT.

3.1 Pre-processing - Embeddings

3.1.1 Provided tokenizer

As the 11,000 aligned samples have been tokenized, we needed to tokenize the samples from the unaligned corpus in a similar manner. Therefore, the first part of the preprocessing was to tokenize the samples in a similar manner as the 11,000 aligned samples to allow one model to process samples from the aligned and the unaligned corpus the same way.

At the same time, to maintain the same representation of corpus, punctuation and capitalization are removed from English unaligned corpus. For french language, it is more complicated. As the aligned french corpus contains punctuation and are cased. So the final predictions from our model should be able to generate the same format of text. Considering the source language doesn't have punctuation and capitalization. It is a possibility to have a translation model focused on finding the mapping on tokens and to leave the grammar correction to an auxiliary model which focus on formatting the target sentences. Our objective is not only to translated but also to obtain a grammatically correct sentence in French with punctuation even if the English sentence does not contain any punctuation information. We explored this approach of adding the punctuation as a post-processing step, but, at the end, we went with models doing both translation and punctuation.

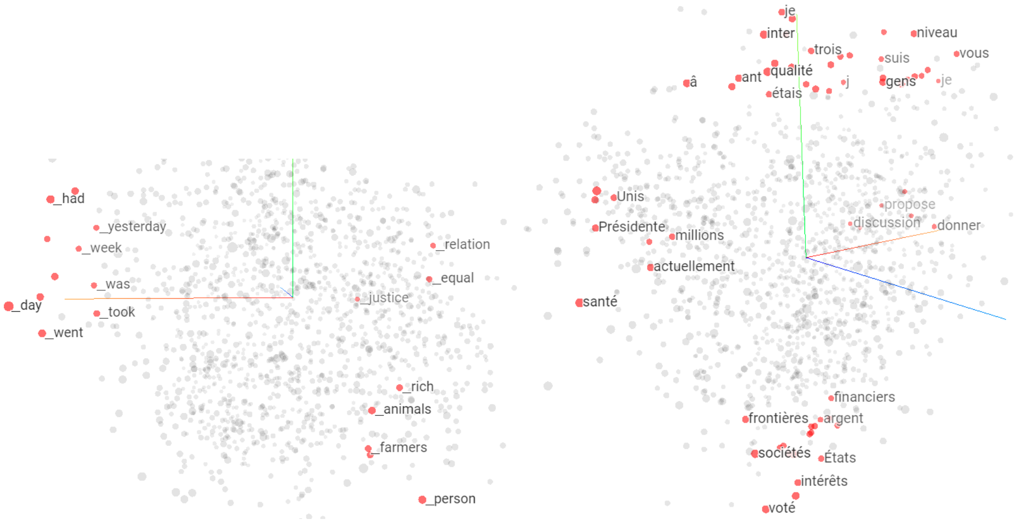
3.1.2 Byte-pair encoding

Once the pre-tokenization was completed, we applied SentencePiece which creates subword units using the byte-pair encoding from Sennrich et al. [2016]. SentencePiece is an unsupervised text tokenizer and detokenizer mainly for Neural Network-based text generation systems where the vocabulary size is predetermined prior to the neural model training. SentencePiece is language independent, therefore it can be used for both English and French samples. Byte-pair encoding split words into sub-words and takes into account the frequency of the sub-words. It provides a balance between character and word tokenization which helps eliminate the out-of-vocabulary (OOV) problem.

3.1.3 Monolingual word embedding: FastText

By using FastText with skipgram, we learned our word embedding in an unsupervised fashion based both mono-lingual contexts. At this point, both languages had their own word embedding. The word embedding used in our models are learned using the tokenized corpus generated by SentencePiece. The word embedding are actually sub-words representation. The skipgram model learns to predict a target word thanks to a nearby word. On the other hand, the cbow model predicts the target word according to its context. The context is represented as a bag of the words contained in a fixed size window around the target word.

Word embedding do learn interesting structures that can capture analogy and semantic. Thus, a simple way to check the quality of a word vector is to look at its nearest neighbors. This gives an intuition of the type of semantic information the vectors are able to capture. In a similar spirit, one can play around with word analogies. For example, we can see if our model can guess what Paris is to France, and compare it to what Berlin is to Germany. The PCA figures 1a and 1b are plots of word embedding demonstrating that similar words with similar meanings are close in space after a PCA analysis. For example, "day", "week", "yesterday" are clustered together. The monolingual word embeddings are essential to get the cross-lingual embedding for unsupervised neural machine translation.



(a) English 3d PCA analysis of embedding space

(b) French 3d PCA analysis of embedding space

Figure 1: PCA analysis of the top 3 principal components for english and french embedings based on our dataset.

3.1.4 Cross-lingual embedding: Vecmap

According to Lample et al. [2017], a cross-lingual embedding is obtained by independently training embeddings of different languages using monolingual corpus, and then mapping them to a shared space. Once we had the FastText embeddings, we used the Vecmap algorithm from Artetxe et al.

[2018] to map them to a common space, finally obtaining a cross-lingual embedding as needed for one of our semi-unsupervised model.

Embedding has the property that similar words are close to one another. For example: pear, apple, and banana are near each other. Meanwhile, cow, dog, and cat are in another cluster. This property of a nice embedding works the same across languages. Therefore, two monolinguals embedding can have similar topography. In this example, we see that the corresponding words for apple, banana, and pear in Basque are also in a cluster.

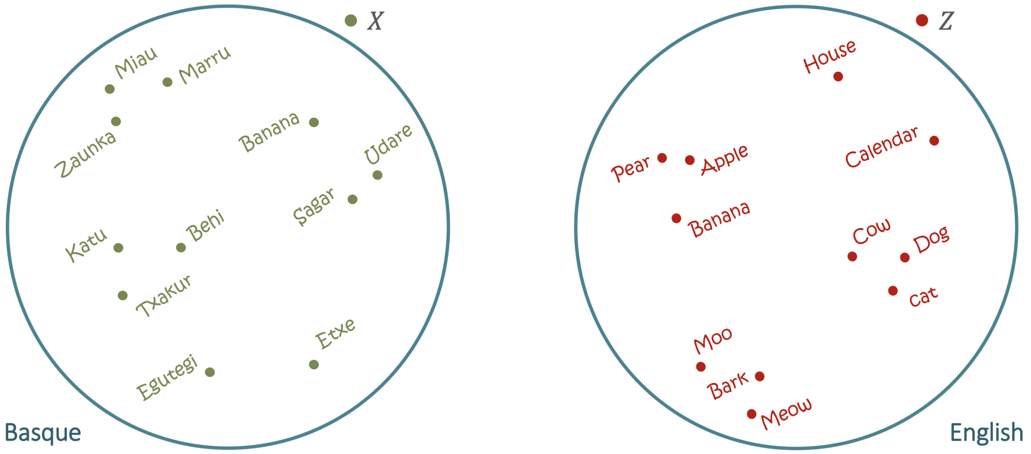


Figure 2: Monolingual embeddings from Artetxe et al. [2018]

Knowing that, it is possible to take advantages of the similarity in both embeddings to align clusters from both embeddings and create a new embedding where the same word in different languages are close to each other.

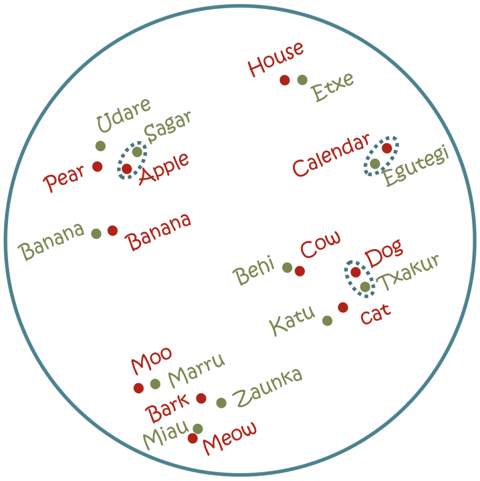


Figure 3: Cross-lingual embedding from Artetxe et al. [2018]

Using the cross-lingual embedding allows to have an embedding of both language in the same space where translation of words from different languages are close together. This mapping to a single space makes it easier to have an encoder that takes sentences from both languages and learns a language independent of representation. The cross-lingual embedding is used in the semi-supervised encoder-decoder model.

4 Translation model

We tried different models on this task with different level of success. We started by creating a baseline model to compare other models' results. Then, we worked on a model that reproduced Artetxe et al. [2017] method and a simple bidirectional LSTM. Note that all models are using the cross entropy loss function during training.

4.1 Bidirectional LSTM (Baseline)

As explained in previous sections, we encountered many issues with our main models which were difficult to fix within the short timeline we had for this experiment. Therefore, we spent some time on a simpler model which could still outperform our baseline. We decided on a seq2seq with attention mechanism.

We have an encoder, a decoder and attention modules. The encoder is implemented with the embedding layer and one bidirectional LSTM layer which has 512 hidden units in each direction. The hidden states from both directions of LSTM are concatenated together and then attention layer attends on both the memory state and carry state.

Regarding the decoder, we just use LSTM and basic encoder. The output tokens which are conditional on the last steps and hidden states. Here, we also tried to apply beam search on decoding. We trained this model using a batch size of 128 due to the limit of GPU memory. Since this model could only be trained using aligned data at first, we obtain results based only on that training portion which can be found in the Results section.

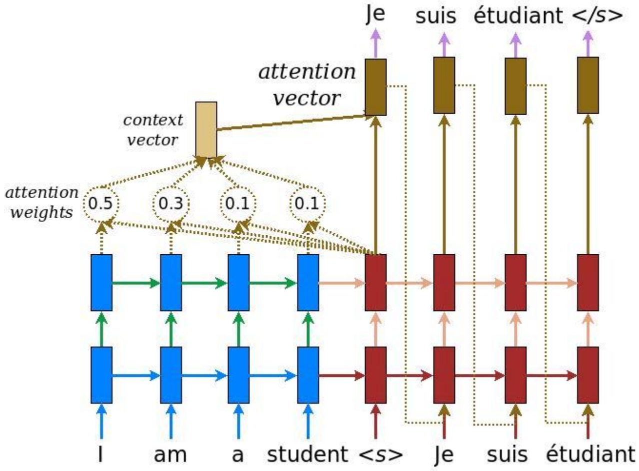


Figure 4: **Input-feeding approach** – Attentional vectors are fed as inputs to the next time steps to inform the model about past alignment decisions.Luong et al. [2015]

4.2 Semi-supervised encoder-decoder model

For this model, we followed the description found in Artetxe et al. [2017]. The encoder contained a two layer bidirectional RNN and the decoder was a two-layer RNN. The RNNs used GRU cells with 600 hidden units and the dimensionality of the embeddings was set to 300. We also included an attention mechanism as described in the paper. The model is constructed in a way that it can deal with translating from English to French but also from French to English. The two tasks share the same encoder but have different decoders. We used the cross-entropy loss function and the Adam optimizer.

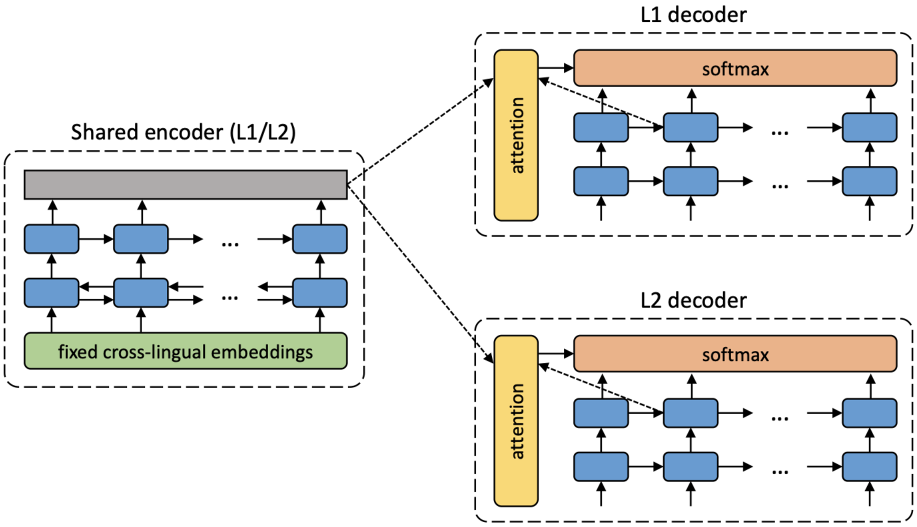


Figure 5: Encoder-Decoder model from Artetxe et al. [2017]

Because we use cross-lingual embedding, the encoder can take sentences from both languages and encodes them into a language independent representation. This representation can then be fed to both decoders. Each decoder translate the representation into their respective language (English and French). For each batch, the sentences are processed by the encoder than translated in both languages. Additionally, during training, to make decoders learn meaningful structural information, noise was added in the decoder's input by swapping the order of some of the words in the sample.

The training consist of five different types of translation. The first four use unaligned data in an unsupervised way and the last one use aligned data. The first two are the translations from a language to the same language (French to French and English to English). It trains the encoder to create a language independent representation and the decoders to be able to translate from the representation to their respective languages. The two next types of training use backtranslation to learn to translate without aligned data. It works by taking a sentence and translating it into the other language. Then, we use the translated sample as training. This way, we generate pseudo aligned data from unaligned data. Backtranslation is done on the fly so that as the model becomes better; the aligned data generated also improves. The last type of translation in the training is the normal English to French translation from the aligned data.

The word embedding was learned during the pre-processing and was kept fixed during the learning and evaluation process. This model has a few advantages: it can use both aligned and unaligned data in a semi-supervised way and there's no need to do post-processing to deal with the punctuation. However, we encountered many problems with this model in Tensorflow which led us to try a different model.

4.3 Bidirectional LSTM with Backtranslation

We use the same model from the Bidirectional LSTM except that instead of training only a English-French model, we also train a French-English model. With the additional model, we can apply the same idea employed in the semi-supervised encoder-decoder model and create pseudo-parallel corpus by using backtranslation on the fly, e.g. translating French samples into English and used those translated sentences to train the English-French model. Therefore, it transforms our previous supervised model to a semi-supervised model and makes use of the unaligned data. Also, because doubling the number of models caused out of memory error, the vocabulary size of the byte-pair encoding was reduced from 10,000 to 5,000.

5 Results

To evaluate our model, we set aside 1,000 samples of our aligned English-French data that we didn't use for training or validation. We used the BLEU score to evaluate the performance of our models. The BLEU (bilingual evaluation understudy) score is often used to evaluate the quality of translated text by NMT models. For this experiment, we used the sacreBLEU implementation from Post [2018]. In this implementation, the BLEU score is based on 4-grams which means that we extract each possible 4-grams from the translated sample and compared them to the target sample. For each 4-gram, if it exists in the target sample, we add 1 and then, we divide by the total number of 4-grams. Then, we calculate the average BLEU score over the 1,000 samples from the test set. The BLEU score is therefore between a range from 0 to 1 and is presented as a percentage in Table 1.

Table 1: Best validation BLEU score of all models

Model	BLEU
Semi-supervised encoder-decoder	18.23
Bidirectional LSTM (Baseline)	9.61
Bidirectional LSTM with Backtranslation	12.73

5.1 Bidirectional LSTM

Bidirectional LSTM obtains a BLEU score of 9.61 using only the 10,000 parallel samples. The training takes approximately 4 hours to complete 30 epochs. Considering the amount of data and low run time, Bidirectional LSTM has a respectable performance.

5.2 Bidirectional LSTM with Backtranslation

By using the unaligned data and transforming the previous model to a semi-supervised model, Bidirectional LSTM with Backtranslation improve the previous BLEU score and obtains 12.73. The score is obtained by doing one epoch of the 474,000 unaligned samples which takes approximately 24 hours. Some samples from the English to French translation can be found in Appendix A. Looking at the samples, we can see that using the byte-pair encoding probably reduces the score because the model occasionally outputs made up words like "mertcher". However, it still helps dealing with the out-of-vocabulary problem.

Compared to the more complex semi-supervise encoder-decoder, there is multiple strategies that could be used to close the gap in performance. Using FastText instead of byte-pair encoding is probably the simplest strategy causing a difference in performance. Other strategies include using a cross-lingual embedding, using a shared encoder instead of two completely separate models, and adding noise in the input by randomly swapping some words in a sample. However, overall, adding the backtranslation on the fly seems like an effective way to improve the performance and, with additional training time and some hyperparameters optimization, the model would probably obtain even better results.

6 Conclusion

The semi-supervised encoder-decoder model was able to obtain the best result with a BLEU score of 18.23. The bi-directional LSTM, also obtained relatively good results but without using the full unaligned data, they were not as high as they could have been. By adding the forward and backward translation methodology to take advantage of the unaligned data, the bidirectional model was able to improve its BLEU score to 12.73 which is a pretty good result when compared to the result obtained by the more complex semi-supervised encoder-decoder.

In the future, it might be useful to start from this model and train it on augmented data following the methodology from Fadaee et al. [2017] or Imankulova et al. [2017]. Or, ideally, we would use those augmented data methodology with the SMT-NMT model proposed by Ren et al. [2019] since it help improve the quality of the pseudo parallel data created from the back-translation process. Also, since we saw that some sample suffered from the byte-pair encoding creating false word as can be seen in

Appendix A, we could take the same model but using the tagging methodology as used by Fadaee et al. [2017] to solve that issue. Considering the limited time available for the experiment, it wasn't possible to make use of those methods but they would probably improve our results. We could also try including more languages in our training so the encoder-decoder model would have less chance of overfitting as this could act similarly to adding noise to the data.

A Samples from bidirectional LSTM with backtranslation

A.1 Good samples

Below are samples where the model was able to keep the overall idea or meaning of the input sentences.

1. **Input:** in fact this crisis is also a crisis for democracy
   Translated: En fait, cette crise est aussi une crise de démocratie.
   Target: En fait, cette crise est aussi la crise de la démocratie.
2. **Input:** it is said that the war will end soon
   Translated: Cela dit que la guerre en fin de la guerre.
   Target: On dit que la guerre s'achèvera bientôt.
3. **Input:** most of you look smart
   Translated: Beaucoup d'entre eux sont intelligentes?
   Target: La plupart d'entre vous a l'air intelligent.
4. **Input:** if it had n't been for the seatbelt i would n't be alive today
   Translated: S'il n'avait pas pu le foncage de sécurité, je ne devais pas être vivant aujourd'hui.
   Target: Sans la ceinture de sécurité, je ne serais pas vivant aujourd'hui.

A.2 Poor samples

Below are samples where the model was not successful at capturing the idea behind the input sentences or where the model created non actual words, due to the use of byte-pair encoding.

1. **Input:** for example any social history will reveal that mutilation and torture were routine forms of criminal punishment the kind of infraction today that would give you a fine in those days would result in your tongue being cut out your ears being cut off you being blinded a hand being chopped off and so on
   Translated: Ainsi, il n'y a pas de symétrie d'une mine et de la torture.
   Target: Par exemple, l'Histoire nous apprend que les mutilations et la torture étaient une punition habituelle pour les criminels. Le genre d'infraction qui aujourd'hui vous donnerait une amende vous causait au Moyen-âge d'avoir la langue ou les oreilles coupées, d'être rendu aveugle, d'avoir une main tranchée, et ainsi de suite.
2. **Input:** we 'll look at sea level rise
   Translated: Nous allons sortir de mertcher.
   Target: Nous regarderons la montée du niveau de la mer.
3. **Input:** i think it has been a successful year
   Translated: J'ai eu une tâche à l'autre.
   Target:Je trouve que cette année a été placée sous le signe de la réussite.
4. **Input:** so the key here is empathy
   Translated: La coupe est un ski.
   Target:L'empathie est ici le point crucial.

References

M. Artetxe, G. Labaka, E. Agirre, and K. Cho. Unsupervised neural machine translation. *CoRR*, abs/1710.11041, 2017. URL `http://arxiv.org/abs/1710.11041`.

M. Artetxe, G. Labaka, and E. Agirre. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, 2018.

E. Cho, J. Niehues, and A. Waibel. Nmt-based segmentation and punctuation insertion for real-time spoken language translation. pages 2645–2649, 08 2017. doi: 10.21437/Interspeech.2017-1320.

M. Fadaee, A. Bisazza, and C. Monz. Data augmentation for low-resource neural machine translation. *CoRR*, abs/1705.00440, 2017. URL `http://arxiv.org/abs/1705.00440`.

T. Ha, J. Niehues, and A. H. Waibel. Toward multilingual neural machine translation with universal encoder and decoder. *CoRR*, abs/1611.04798, 2016. URL `http://arxiv.org/abs/1611.04798`.

A. Imankulova, T. Sato, and M. Komachi. Improving low-resource neural machine translation with filtered pseudo-parallel corpus. In *Proceedings of the 4th Workshop on Asian Translation (WAT2017)*, pages 70–78, Taipei, Taiwan, Nov. 2017. Asian Federation of Natural Language Processing. URL `https://www.aclweb.org/anthology/W17-5704`.

G. Lample, L. Denoyer, and M. Ranzato. Unsupervised machine translation using monolingual corpora only. *CoRR*, abs/1711.00043, 2017. URL `http://arxiv.org/abs/1711.00043`.

M.-T. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

M. Post. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium, Oct. 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-6319. URL `https://www.aclweb.org/anthology/W18-6319`.

S. Ren, Z. Zhang, S. Liu, M. Zhou, and S. Ma. Unsupervised neural machine translation with SMT as posterior regularization. *CoRR*, abs/1901.04112, 2019. URL `http://arxiv.org/abs/1901.04112`.

S. Sen, K. K. Gupta, A. Ekbal, and P. Bhattacharyya. Multilingual unsupervised NMT using shared encoder and language-specific decoders. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3083–3089, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1297. URL `https://www.aclweb.org/anthology/P19-1297`.

R. Sennrich, B. Haddow, and A. Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1162. URL `https://www.aclweb.org/anthology/P16-1162`.

O. Tilk and T. Alumäe. Bidirectional recurrent neural network with attention mechanism for punctuation restoration. pages 3047–3051, 09 2016. doi: 10.21437/Interspeech.2016-1517.