# Natural Language Processing through Neural Machine Translation

**COMP550 – Final Project – Fall 2017**

Philippe Lacaille
Lucas Pagé-Caccia

## Abstract

As a project for the Natural Language Processing (COMP550) class, we propose to explore the translation of natural language descriptions of images from the WMT17 Multimodal Machine Translation task. This being a shared task with published results from the community, we show how neural machine translation can perform well with known architectures without any over engineering specific to this task. We further explore how adding knowledge such as image features and pre-trained embeddings affects performance, as well as jointly training the encoder for multiple languages. All source code, data and best model outputs are available at the following repository: `https://github.com/placaille/nmt-comp550`.

## 1 Introduction

Translating from a language to another requires a deep understanding of both the source and target language from the morphological to the pragmatic level. This is known to be a difficult task; as humans, we can relate by experience to learning a new language.

Using machine learning algorithms with high capacity recurrent neural networks, known as neural machine translation, has been seen as new successful approach in recent years. These models have the advantage to be trained on very large corpora in order to *discover* the relationships and dependencies between the two languages automatically as part of their training procedure.

However, in this project, we focus on a shared task consisting of only 29,000 data points in the training corpus, the WMT17 Multimodal Machine Translation task (Elliott et al., 2016). The 2017 shared task is the second edition, with French being added to English and German translation. The first task of this challenge that we selected for this project consists of having English descriptions of images translated to either French or German, with the possibility of using the raw images or extracted features. Participants to this shared task submitted their predictions on the test set, only then to see the rankings of performance based on each submission's model. More details on the findings of the shared task's results and models will be discussed later.

A second task is also available under the same contest, where participants are to generate directly the description solely from the image. For each image (same as previous task) five sentences in each language are available for training. Even though this second task does contain a natural language processing aspect, we opted to explore the first task described as it shared more grounds with the contents/goals of the class.

We propose to show how widespread recurrent neural network architectures compare to models that performed well in the shared task. Briefly, we start by using only the text data with models in their simplified form such as LSTM and GRU units, with enhancements such as attention and bidirectional layers. Our hope is to notice a high performance from the *basic architectures* when compared to more customized models that participants submitted. In addition to using deep recurrent networks

in their simplistic form, we explore how enhancing the knowledge available helps performance. This is done through incorporating the image features to the model and as well as by using pre-trained word embedding in the encoder network. Finally, we test a new approach of mixing training between English to French and English to German to see if it enhances the encoding performance of the English model. We hope to show that this approach will increase the monolingual translation performance of English to French.

## 2 Related work

Ten teams participated in the first edition of this multimodal translation tasks. All successful approaches combined a powerful (LSTM/GRU) encoder-decoder network equipped with an attention mechanism (Bahdanau et al., 2014), (Luong et al., 2015) over textual features. The proposed models differ mainly in *how they ground the translation on images*. Three visual inputs are available to participants, refer to 3.1. Simpler methods initialize the decoder's hidden state by a simple concatenation the encoder's last hidden state and the global pooled features. Similar in flavor, other teams did this concatenation for every decoding timestep. More sophisticated methods used variant of attention model for visual features (Xu et al., 2015) on the intermediate VGG representation.

The second edition of this task gathered nine teams who submitted 19 systems. The building blocks for each of these systems are the same as for the first edition, but are composed differently. The second best team used a hierarchical-based translation (HBT) system build using Moses (Koehn et al., 2007). They then leverage an attentional encoder-decoder (Bahdanau et al., 2014) to rerank the n-best output of the HBT system. They don't condition on the image features directly, but instead use image retrieval methods to fetch target descriptions of similar images. This approach yields a 55.3 BLEU score, compared to the text-only baseline of 44.3. The winning team, scoring a BLEU of 55.9, (Caglayan et al., 2017) went for a simple, elegant approach, and did not use an additional attention mechanism: they condition the target embedding with the global pool features, i.e. $y_j = y_j \odot tanh(W_{img} \cdot V_{pool})$. Their approach also included layer normalization (Ba et al., 2016) and a two layer decoder.

## 3 Method

### 3.1 Data and experimental setup

The specific task tackled in this project is translating from English to French as opposed to German. The dataset provided contained a training set of 29,000 image descriptions, an additional 1,014 in the validation set as well as the 2016 test set with 1,000 data points. In addition, two 2017 test sets (Elliott et al., 2017) were available, one being an *in-domain* evaluation from Flickr, the other being an *out-of-domain* from the MSCOCO dataset. We selected the *in-domain* dataset, which had also 1,000 data points to evaluate.

Since there was no French in the 2016 version of this task, only results on the 2017 test set were reported by the organizers collection of results. In this project, we consider both 2016 and 2017 test sets as true test sets, i.e. no decision is made based on results and their sole purpose is to give an estimation of generalized performance of the models.

By considering only the training set, we obtain a dictionary of 9,935 entries for English and 11,618 entries for French (both of these vocabularies include start-of-sentence, padding, unknown and end-of-sentence tokens). These tokens were obtained by lowercasing all sentences and normalizing punctuation. As for the validation and test sets, we replace all tokens from the source sentences not part of the source language dictionary by the unknown token.

Given a tokenized input sentence in the source language, we therefore aim to generate a translated version of it into the target language. To do so, all of our models utilize an encoder-decoder structure, for more details on the models see subsection 3.2. At each step (token) of the generation process, our model generates a probabilistic distribution over all of the target language dictionary, conditioned on the input sentence in the source language and all other previously predicted words in this sentence. Under the mathematical formulation, the conditional distribution

$$P(y_t|y_{t-1}, y_{t-2}, \cdots, y_1, \mathbf{C})$$

where $\mathbf{C}$ denotes the context and $y_k$ denotes the $k^{th}$ token. In more practical terms, the context is the

fixed size embedding that is provided by the encoder network to the decoder network. It therefore represents all the information of the input sentence compressed into a vector, making the generation process a conditional process.

In the training procedure, we use a hyper-parameter to define the probability of teacher forcing for the full sequence. With a probability $p$, for all time steps $t \in \{1, \cdots, T\}$ of a sequence, the previously generated token $y_{t-1}$ is replaced with the true $t^{th}$ token from the target sentence. Otherwise, the input token to the next step is the one that is most likely out of the current distribution. Using the sequence of conditional distributions and the target sentence, the networks are trained end-to-end using the categorical cross-entropy loss with a variant of gradient descent, the Adam optimization algorithm. Multiple values of learning-rate between $10^0$ and $10^{-4}$ were explored and was annealed by a factor of four if no improvement on the validation loss is seen between epochs.

For the generative process, we use a beam search of size 15 to find the best sequence of tokens to predict. This ensures that currently suboptimal but later optimal selections are considered rather than opting for the greedy approach at each step. To evaluate the predicted sentences, we compare them with the gold translations, which are processed the same way as the predictions, and score them using the BLEU score. The BLEU score is a measure of overlap between predictions and reference corpus, by counting the frequency of n-grams of the reference sentence that occur in the predicted sentences.

### 3.2 Model architecture

In order to test widespread recurrent neural network architecture associated with neural machine translation, we built our encoder and decoder as either LSTM or GRU units. Both of these models have shown to perform with long-term dependencies by incorporating gating units to filter out or add new information to the cell/hidden state. Note here we do not go in further details as to describe the LSTM and GRU units, you can refer to (Hochreiter and Schmidhuber, 1997) and (Cho et al., 2014). As an input to these units, we use trainable word embeddings as a mapping from the vocabulary size to a fixed size vector. As part of hyper-parameter searching, we explored LSTM and GRU networks up to two layers, as well as unidirectional or bidirectional encoder while keeping the decoder unidirectional. The number of hidden units and the size of the embedding layers both varied between 100 and 700.

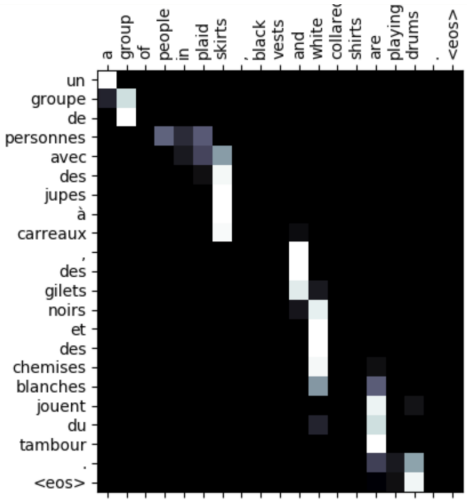


**Figure 1:** Attention weights at each step of the generation process for a sample sentence from the 2016 test set.

Sequence to sequence models like the ones we use compress the input sequence into a fixed-size embedding (known as the context) and this can limit the information passed along to the decoder. Indeed, the context is used as the initial hidden state of the decoder. Attention is a mechanism that aims to increase the information available to the decoder by allowing it to access the series of hidden states that were created during the encoding of the input sequence. This means that at each step of the decoding, the model can now weight differently the encoder hidden states and this can be seen as a way of focusing by the decoder. While many attention mechanisms exist in the community, we opted to implement the one described in (Luong et al., 2015). Refer to Figure 1 for an example of the attention weights at each step.

In all architectures explored, we decided not to consider specific methods to deal with the unknown tokens. Dealing with a fix vocabulary and unknown tokens is a research area in itself, it is also interesting to consider how well models can perform without any consideration of this.

All our implementation was done using PyTorch `https://github.com/pytorch/pytorch`.

### 3.3 Enhancing knowledge

To enhance knowledge available to the models, we first opted for adding features of the images. The dataset also makes available visual features from a pre-trained ResNet-50 model as full or average pooled features; we opted for the latter. In order to incorporate the image features of size 2,048, we concatenate it with the encoder's final output (the context **C**) and pass it through a dense layer and obtain a revised context **C'** of the same size then fed to the decoder. By doing so, we hope that the information in the image features will help the decoder grasp what the description is about and therefore translate the sentence accordingly.

Another enhancement is replacing our embedding layer with pre-trained word embeddings. These pre-trained word embeddings are trained on much larger corpus than the one we are working with in order to offer a mapping from words to a subspace that is considerably smaller than the vocabulary space. This allows for words that are used in similar context to be close in the embedding space ($\mathbb{R}^{300}$ for the model used). By using these, hopefully, our model would then exploit similarity between words that maybe it didn't see much in the training set. In our implementation, we use Google's Word2Vec model (Mikolov et al., 2013), more information can found on our repository. We use only pre-trained word embeddings for the English encoder and we tested by either forcing the embeddings to stay the same or allow for further training through the previously described training process.

Additionally we explore adding a second language to translate. The translation from English to German is done with the same encoder as the English to French translation, while both target languages have their decoders. We train both networks simultaneously, and we pass the loss signal onto the unique encoder appropriately. We hope the shared encoder would learn a representation invariant to the output language, also serving as a regularization mechanism to the model.

## 4 Results

We tested multiple configurations on the text-only corpus and the most successful ones were selected based on the performance on the validation set as described above. To restrict the number of models evaluated, we only searched the knowledge enhancement variants within a limited set of the successful text-only configurations. Refer to Table 1 for BLEU scores under few of the most successful configurations.

| **Model variant** | **Valid** | **T2016** | **T2017** |
|---|---|---|---|
| text-only | | | |
| - 600h-300e-0.3tf | 49.5 | 51.1 | 42.2 |
| - 500h-400e-0.2tf | 47.7 | 49.4 | 41.5 |
| image features | | | |
| - 500h-300e-0.3tf | 50.0 | 51.4 | 43.1 |
| - 600h-250e-0.3tf | 49.2 | 51.6 | 42.8 |
| word embeddings | | | |
| - 600h-300e-0.3tf-train | 29.3 | 29.7 | 22.7 |
| - 600h-300e-0.3tf-notrain | 29.2 | 30.1 | 23.2 |
| two language decoder | | | |
| - 600h-400e-0.3tf | 48.8 | 50.7 | 42.0 |
| - 600h-250e-0.3tf | 48.3 | 49.9 | 42.2 |

**Table 1:** BLEU scores for the best model configurations on the validation, 2016 and 2017 test sets. All the models in the table used LSTM units and dropout with $p = 0.5$. A model described as Xh-Ye-Ztf represents a model of X hidden units, embedding of size Y and teacher forcing probability Z.

Using only the text corpus, we can surprisingly almost match the results listed in Section 2 with BLEU scores of 51.1/42.2 on the 2016/2017 test datasets. We can notice out of the sample predictions in Table 2 that the second sample is a perfectly valid translation, that probably scored lower on the BLEU score due to the fact it wrote *black baseball cap* as *casquette de baseball noire* rather than the gold version of *casquette noire*. This is an example where BLEU score was wrong just because the given gold translation wasn't perfect.

As expected, allowing the model to condition on the image features resulted in a noticeable quantitative performance increase. Indeed, we were able to boost our BLEU scores by $\approx 0.5$, outperforming all our other models. We believe that the image features allow the model to build a more global representation of the input. However, a qualitative analysis of the output sentences with respect to the best text-only model shows no significant difference. The only noticeable difference are for sentences with colors in it where it relieved the model from having to learn these features from text, allowing it to extract other relevant information for better output.

| |
|---|
| **English** |
| 1- people sitting in a circle outside a large building .<br>2- a woman in a gray sweater and black baseball cap is standing in line at a shop . |
| **French** |
| 1- des gens assis en cercle devant un grand bâtiment .<br>2- une femme avec un pull gris et une casquette noire est debout et fait la queue dans un magasin . |
| **Best text-only model** |
| 1- des gens assis en cercle devant un grand bâtiment .<br>2- une femme avec un pull gris et une casquette de baseball noire est debout en ligne dans un magasin . |
| **Best image-features model** |
| 1- des gens assis en cercle devant un grand bâtiment .<br>2- une femme avec un pull gris et une casquette de baseball noir est debout au ligne d'un magasin . |
| **Best word embeddings model** |
| 1- des gens assis dans un bâtiment devant un grand bâtiment .<br>2- une femme avec un t shirt gris et un pantalon noir se tient debout dans une un de . . . . |
| **Best two language decoder model** |
| 1- des gens assis dans cercle devant un grand bâtiment .<br>2- une femme avec un pull gris et une casquette de baseball noir est debout en ligne dans un magasin . |

**Table 2:** Sentences from 2016 test set predictions along with the corresponding gold translation.

Surprisingly, the pre-trained word embeddings did not seem to help the model but rather confuse it as it evidently lowered performance. In addition, for different configurations, the BLEU score was very similar even by allowing the embeddings to continue training after being initialized with the pre-trained model. Initializing the embedding values to the Word2Vec's may have caused the model to be stuck in a local minimum that even allowing further training could not help. Interestingly, from the sample sentences in Table 2 we can notice concepts used in similar context being used as a consequence of the pre-trained word embeddings (referred to *pantalons* as opposed to *casquette*).

The last model variant we tried was to train a model with two language decoders simultaneously. When compared to text-only results we do not see an improvement of performance. We hypothesize this is caused by the two datasets having all the same sentences. Having different a corpus for the English to German, we think it could have made a positive impact on performance.

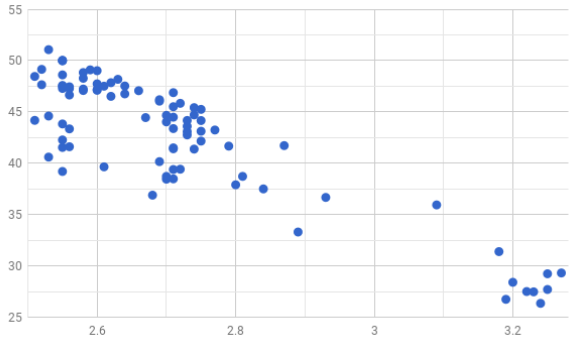


**Figure 2:** BLEU score on the y-axis and validation loss x-axis on the validation set for different model configurations

## 5 Discussion and conclusion

Interestingly, we noticed that having a lower validation loss did not necessarily result in a higher BLEU score. We illustrate this in Figure 2 where we tally most of our validation loss and BLEU scores. This makes us question the use of the categorical loss as a surrogate of the true objective. Some thoughts could be made into investigating the training objective.

We further noticed a large difference in performance between the 2017 test set compared to validation and 2016 test set $\approx$ 7 points. We suspect this is caused by the small dataset size, leading to high variance. Regarding the unknown tokens, we tested performance by ignoring the sentences that contained such a token and we noticed an increase in performance ranging between 2 and 4 BLEU points. This confirmed that basic RNN models were pretty robust even though no special treatment of these tokens was considered. Using the image features differently seemed to be key to achieve higher performance. Some other approach to explore further could be to incorporate them at each step of the decoder, or even at each step of the encoder, or both.

As a general takeaway, we are satisfied to see that by using only the text data and some basic architectures, the models tend to perform very well. It was further encouraging to see that, like humans, our models performed better by leveraging visual content.

## Statement of contributions

Both Lucas Pagé-Caccia and Philippe Lacaille were responsible for the design, the writing and the implementation of the project. All work was done cooperatively as a team, Philippe was more involved in the core implementation of the text-only and word embeddings variant, while Lucas was more involved in implementing the attention mechanism and variants of the second language decoder and image features.

## References


[Ba et al.2016] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

[Bahdanau et al.2014] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

[Caglayan et al.2017] Ozan Caglayan, Walid Aransa, Adrien Bardet, Mercedes García-Martínez, Fethi Bougares, Loïc Barrault, Marc Masana, Luis Herranz, and Joost van de Weijer. 2017. Lium-cvc submissions for wmt17 multimodal translation task. *arXiv preprint arXiv:1707.04481*.

[Cho et al.2014] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

[Elliott et al.2016] D. Elliott, S. Frank, K. Sima'an, and L. Specia. 2016. Multi30k: Multilingual english-german image descriptions. In *Proceedings of the 5th Workshop on Vision and Language*, pages 70–74.

[Elliott et al.2017] Desmond Elliott, Stella Frank, Loïc Barrault, Fethi Bougares, and Lucia Specia. 2017. Findings of the Second Shared Task on Multimodal Machine Translation and Multilingual Image Description. In *Proceedings of the Second Conference on Machine Translation*, Copenhagen, Denmark, September.

[Hochreiter and Schmidhuber1997] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

[Koehn et al.2007] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

[Luong et al.2015] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

[Mikolov et al.2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

[Xu et al.2015] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*, pages 2048–2057.