# Neural Machine Translation: Vietnamese → English

**Anu-Ujin Gerelt-Od, Lee Kho, Anhthy Ngo, Parthvi Shah**

Center for Data Science, New York University
{ago265,ltk224,an3056,pss434}@nyu.edu

## Abstract

Neural machine translation is one of the most challenging research areas in natural language processing. The performance of NMT systems are highly sensitive to the amount of available training data. For this reason, low resource language pairs such as English-Vietnamese often suffer from lower performance. In the scope of this project, we work on the IWSLT'15 English-Vietnamese translation task, experiment with different encoder-decoder architectures, and perform hyperparameter tuning to maximize NMT model performance. We evaluate our model through two methods: (1) by replacing out-of-vocabulary words with ⟨UNK⟩ tokens and (2) leaving the original sentence unchanged. With the attention mechanism proposed by Bahdanau et al. (2014), we achieved final validation BLEU scores of 26.56 and 25.37 with methods 1 and 2, respectively.

## 1 Introduction

Neural machine translation (NMT) involves translating a source sentence $F = f_1, \ldots, f_N$ into a target sentence $E = e_1, \ldots e_M$, where $N$ and $M$ can be of variable length. Thus, we define any translation system as:

$$\hat{E} = \text{mt}(F)$$

which returns a translation hypothesis $\hat{E}$ given a source sentence $F$ as input. NMT are deemed to be one of the most popular yet challenging NLP tasks mostly due to the fluidity of natural languages that cause problems with capturing grammatical structures and nuances. There have been impressive machine translation results for language pairs such as English-French (Bahdanau et al., 2014) or English-Chinese (Cheng et al., 2018), but English-Vietnamese translations face challenges that mainly stem from low-resource conditions. In this paper, we seek to improve these translation efforts for Vietnamese → English by training the IWSLT'15 English-Vietnamese dataset with a sequence-to-sequence framework using LSTM and GRU architectures.

**Recurrent Neural Networks** Machine translation tasks involve variable length sequences which can be modeled by recurrent neural networks (RNNs). Unlike feed-forward networks, which only allow signals to travel in one direction, RNNs allow outputs to be fed back into the network as inputs using loops. Bengio et al. (1994) have shown that vanilla RNNs fail to capture long-term dependencies due to exploding (less likely) or vanishing gradients (more likely) during gradient-based learning. Special types of RNNs have been introduced, such as Long Short-Term Memory (LSTM) proposed by Hochreiter and Schmidhuber (1997) and Gated Recurrent Units (GRUs) proposed by Cho et al. (2014) to overcome these long-term dependencies.

**Sequence to Sequence Learning** As mentioned, general feed forward networks do not have the flexibility to map sequences to sequences. The seq2seq model was introduced by Sutskever et al. (2014) and aims to transform an input source sequence to an output target sequence, where both sequences can be of arbitrary lengths. The seq2seq model consists of an encoder which processes and compresses an input sequence into a fixed length context vector, and a decoder which extracts the output sequence from that context vector. The encoder and decoder networks are RNNs, generally LSTMs or GRUs, which are jointly trained to maximize the probability of a correct translation given a source sequence.

**Attention Mechanism** The major pitfall of the basic encoder-decoder architecture (known as "the

| RNN | Training | RNN Layers | Optimizer | Beam Size | BLEU (UNK) | BLEU (No UNK) |
|---|---|---|---|---|---|---|
| LSTM | w/o Attn | 1 | SGD | 3 | 10.36 | 9.87 |
| LSTM | w/o Attn | 2 | SGD | 3 | 11.21 | 10.70 |
| LSTM | w/o Attn | 3 | SGD | 3 | 11.60 | 11.04 |
| LSTM | w/o Attn | 1 | SGD | 5 | 9.91 | 9.42 |
| LSTM | w/o Attn | 2 | SGD | 5 | 11.60 | 11.12 |
| LSTM | w/o Attn | 3 | SGD | 5 | 11.98 | 11.35 |
| GRU | w/o Attn | 1 | SGD | 3 | 9.91 | 9.42 |
| GRU | w/o Attn | 2 | SGD | 3 | 11.07 | 11.64 |
| GRU | w/o Attn | 3 | SGD | 3 | 12.29 | 11.63 |
| GRU | w/o Attn | 1 | SGD | 5 | 10.05 | 9.62 |
| GRU | w/o Attn | 2 | SGD | 5 | 11.92 | 11.29 |
| **GRU** | **w/o Attn** | **3** | **SGD** | **5** | **12.10** | **11.48** |
| LSTM | Attn | 1 | SGD | 3 | 24.61 | 23.43 |
| LSTM | Attn | 2 | SGD | 3 | 25.73 | 24.54 |
| LSTM | Attn | 3 | SGD | 3 | 26.53 | 25.35 |
| LSTM | Attn | 1 | SGD | 5 | 24.80 | 23.61 |
| LSTM | Attn | 2 | SGD | 5 | 25.73 | 24.40 |
| **LSTM** | **Attn** | **3** | **SGD** | **5** | **26.56** | **25.37** |
| GRU | Attn | 1 | SGD | 3 | 23.50 | 22.32 |
| GRU | Attn | 2 | SGD | 3 | 24.02 | 22.80 |
| GRU | Attn | 3 | SGD | 3 | 24.65 | 23.35 |
| GRU | Attn | 1 | SGD | 5 | 23.37 | 22.23 |
| GRU | Attn | 2 | SGD | 5 | 23.50 | 22.80 |
| GRU | Attn | 3 | SGD | 5 | 24.62 | 23.30 |

Table 1: Hyperparameter search shows Attention dramatically improves validation BLEU scores. Best models for Attention and w/o Attention are bolded. GRU and LSTM RNNs show similar results with LSTM having slight edge in performance, on average.

bottleneck problem") stems from the fixed length context vector. The context vector is a numerical summary of an input sequence and it would be unreasonable to expect that this one vector representation would be able to decode longer input sequences, and this would eventually lead to catastrophic forgetting. The attention mechanism, which automatically (soft-) searches for parts of the source sequence that are relevant to predicting the target sequences, overcomes this issue (Bahdanau et al., 2014).

## 2 Data

We trained our model on the IWSLT'15 English-Vietnamese Dataset - a parallel English-Vietnamese corpus from The Stanford NLP Group[1] that was compiled through the transcription and translation of TED and TEDx talks, i.e. public speeches covering many different topics. The training data contains 133,317 sentence pairs, while the validation and test sets consist of 1,268 and 1,553 pairs, respectively.

[1] https://nlp.stanford.edu/projects/nmt/

## 3 Related Work

Koehn and Knowles (2017) found that NMT performs poorly without a large training corpora. The lack of a well-sized parallel corpora for English-Vietnamese tasks have provided practical challenges to neural based approaches. However, there has been a few experimental projects that have used the IWSLT'15 dataset for similar machine translation tasks.

**Vietnamese Sequence Learning** Luong and Manning (2015) developed a baseline NMT model that uses sequence-to-sequence RNNs that are trained end-to-end on a large corpora. It is based on the encoder-decoder framework that uses conditional probability to translate the target words as they come up. When trained with a small English-Vietnamese dataset, the model performed quite well despite having only a few LSTM layers. We will compare our BLEU score with this model, which achieved a score of 26.4.

**Further Vietnamese NMT** Phan-Vu et al. (2018) helped improve English-Vietnamese translations by building the largest open Vietnamese-English corpus and conducted extensive experiments for BLEU score optimization for low-

resource language pairs. They experimented with and combined various NMT architectures, including RNN, Transformer, and Convolutional sequence-to-sequence (ConvS2S) with tuned hyperparameters and achieved a BLEU score of 40.01 and 35.81 for the English-Vietnamese and Vietnamese-English translations, respectively.

## 4 Models

**Encoder-Decoder RNN** For our NMT architecture, we used an encoder-decoder RNN model and experimented with both bi-directional LSTMs and GRUs for our encoder block. LSTMs are composed of a cell state, an input gate, a forget gate, and an output gate (Hochreiter and Schmidhuber, 1997). The input gate regulates how much of the new cell state to keep, the forget gate regulates how much of the existing memory to forget, and the output gate regulates how much of the cell state should be exposed to the next layers of the network.

Unlike LSTMs, GRUs do not have cell states, but instead use hidden states to transfer information. GRUs have two gates – an update gate and a reset gate (Cho et al., 2014). The update gate (similarly to the forget and input gates of an LSTM) regulates which information to discard and which additional information to add, while the reset gate regulates how much prior information to forget. GRUs have been shown to produce similar performance with LSTMs and, due to their less complex internal structure, are less computationally expensive (Chung et al., 2014).

**Encoder-Decoder RNN with Attention** To boost performance, we also implemented an additive attention module (Bahdanau et al., 2014) to incorporate into our decoder block. Attention mechanisms are designed to allow the decoder to directly access all of the encoder's hidden states, as opposed to only accessing the context vector output of the encoder. This results in shorter dependencies in our computation graph as we have a more direct connection between each of the tokens that we are decoding and each of the input tokens.

With an attention mechanism, the decoder takes in three inputs at each timestep $t$: the previous timestep's hidden state $s_{t-1}$, the previous output of the decoder $y_{t-1}$, and the context vector $c_t$, which is a weighted average of the encoder's hidden states $h_i$ for $i = 1, 2, ..., n$ (since the encoder is a bidirectional LSTM, each $h_i$ is a concatenation of both the forward and backward hidden state vectors). The attention weights of $c_t$ (denoted as $\alpha_i$ for $i = 1, 2, ..., n$) are the outputs of a softmax layer performed on a predetermined scoring function that measures the alignment between each $h_i$ and $s_{t-1}$.

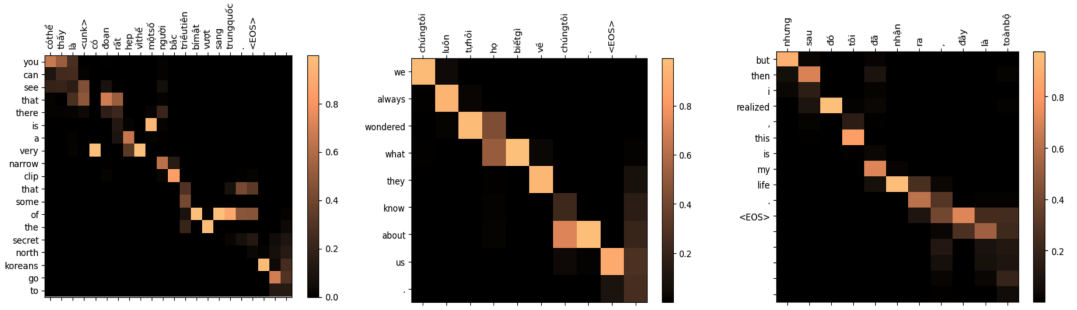
## 5 Experiments

**Methodology** The NMT models were trained for 15 epochs each on the NYU's High Performance Computing (HPC) Prince Cluster. The initial learning rate is set to $0.25$ for the SGD optimizer and $1e$-$5$ for Adam, with a minimum threshold of $1e$-$4$ to reduce the learning rate when the validation loss plateaus by a factor of $0.5$.

The performance of the trained models are measured by their BLEU scores (Papineni et al.), and evaluated on two differently tokenized versions of the validation and test sets. In method 1: BLEU (UNK), all out-of-vocabulary (OOV) words are replaced with an $\langle$UNK$\rangle$ token, whereas in method 2: BLEU (No UNK), we keep the original reference sequence as is. With method 1, we are able to predict more n-grams correctly and thus generate a slightly higher BLEU score. This is because when we encounter an OOV word in the test set, we will predict $\langle$UNK$\rangle$ rather than predicting a token highly likely to be a mismatch.

**Evaluation** For our models, the prediction required is a sequence of words. Thus the model first outputs a probability distribution over each word in the target vocabulary for each word in the output sequence, then the decoder is left to transform these probabilities to a final sequence of words. For evaluation of these decoded sentences, we aim to avoid sub-optimal translations. Ideally, we want to select the target word with maximum probability based on the input sentence at each time step, but choosing *only one* best candidate (Greedy Search) at each time step may not be suitable, because when constructing the full sentence, some of the times we may need other candidates from previous time steps to have a more accurate translation.

To overcome sub-optimal translations, we implement beam search, which selects more than one alternative (based on the set beam size) for the best target word at a given time step based on conditional probability. After selecting $b$ best candidates at each time step, we make a new vocabulary which consists of all the previous candidates,



**Source:** cóthể thấy là <unk> có đoạn rất hẹp vìthế mộtsố người bắc triềutiên bímật vượt sang trungquốc .
**Predicted:** you can see that there is a very narrow clip that some of the secret north koreans go to china .
**Actual:** As you can see , the river can be very narrow at certain points , allowing North Koreans to secretly cross .

**Source:** chúngtôi luôn tựhỏi họ biếtgì về chúngtôi .
**Predicted:** we always wondered what they know about us .
**Actual:** We always wondered what they knew about us .

**Source:** nhưng sau đó tôi đã nhận ra , đây là toànbộ cuộcsống của mình .
**Predicted:** but then i realized , this is my life .
**Actual:** But then after a while I realized , this is my life .

Figure 1: First figure shows the complications with ⟨UNK⟩ tokenized examples. Second figure shows small grammatical issues that arise from our translation model. Third figure is an example of a good machine translation.

as well as the current candidates. Next, we keep on predicting one word at a time until the beam search picks ⟨EOS⟩ as the final token. A higher value of beam size would give a more accurate translation, but it is computationally more expensive. Hence, we keep our beam sizes at 3 and 5.

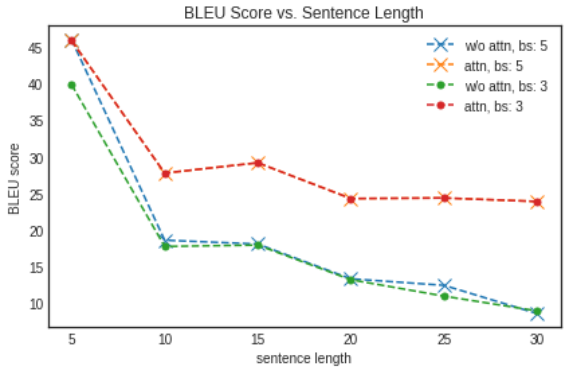



Figure 2: Increasing beam size from 3 → 5 show marginal improvements to BLEU scores. Attention mechanism outperforms w/o Attention through all sentence lengths.

**Results** Our model with the best results was structured with a bi-LSTM encoder (3 layers, hidden dimension of 512) and a LSTM decoder (3 layers, hidden dimension of 1024, with attention) that was evaluated with a validation beam size of 5. The model resulted in a validation BLEU score of 26.56 (UNK) and 25.37 (No UNK). From Figure 2, we can see that increasing beam size shows only marginal improvements, but adding in the attention mechanism gives a significant boost to the validation BLEU scores.

## 6 Error Analysis

Figure 1 highlights some sample translations from our machine translation model. In general, OOV words in the source sentence are problematic for our model. This arises when a word in the source language doesn't meet the minimum count of 5, or does not exist in the target vocabulary. Because the word is not learned by the model, the decoder will fail to predict it or will miss out on some information from the source sentence. Additionally, our model fails to recognize small grammatical nuances, but the main meaning is preserved. From Figure 2, it is evident that our model performs better on short sentences (with a maximum BLEU score of around 5 words per sentence) and struggles more on longer sentences.

## 7 Future Work

NMT has shown SOTA results with convolutional sequence-to-sequence and transformer based models. Phan-Vu et al. (2018) experimented with ensemble methods, combining pairs of techniques mentioned above which improved their BLEU score significantly. We would like to extend our research by experimenting with such ensembles of NNs and performing additional hyperparameter tuning such as exploring additional beam sizes. Additionally, we hope to try a phrase-based segmenter approach (Luong and Manning, 2015), as well as BPE tokenization to verify if the decoder performs well on logo-graphic languages such as Vietnamese and Chinese.

## References


Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate.

Y. Bengio, P. Simard, and P. Frasconi. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166.

Yong Cheng, Zhaopeng Tu, Fandong Meng, Junjie Zhai, and Yang Liu. 2018. Towards robust neural machine translation.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

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

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. *CoRR*, abs/1706.03872.

Minh-Thang Luong and Christopher D. Manning. 2015. Stanford neural machine translation systems for spoken language domains.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. *ACL*, abs/P02-1040.

Hong-Hai Phan-Vu, Viet-Trung Tran, Van-Nam Nguyen, Hoang-Vu Dang, and Phan-Thuan Do. 2018. Neural machine translation between vietnamese and english: an empirical study.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. *CoRR*, abs/1409.3215.