# A Sequence-to-Sequence Model with Attention for Korean-to-English Translation

HyunJae Lee

June 4, 2025

**Abstract**

This paper details the development of a Neural Machine Translation (NMT) model for translating text from Korean to English. The model employs a sequence-to-sequence (Seq2Seq) architecture, a powerful framework for handling variable-length input and output sequences. The core components of the model are an encoder, which processes the Korean source sentence, and a decoder, which generates the corresponding English translation. To address the challenge of handling long-range dependencies and to improve translation quality, a Bahdanau attention mechanism is integrated into the architecture. The model is trained on a parallel corpus of Korean and English sentences. We describe the data preprocessing pipeline, the mathematical foundations of the encoder-decoder and attention mechanisms, and the training procedure. The results demonstrate the model's ability to generate coherent, albeit simple, translations, confirming the effectiveness of the attention-based Seq2Seq approach for this task.

## 1 Introduction

Machine Translation (MT), the task of automatically converting text from a source language to a target language, is a cornerstone of natural language processing (NLP). The advent of deep learning has revolutionized this field, leading to the development of Neural Machine Translation (NMT), which has significantly surpassed traditional Statistical Machine Translation (SMT) in performance.

### 1.1 Background

Early NMT models were pioneered by Sutskever et al. (2014), who introduced the sequence-to-sequence (Seq2Seq) framework. This framework consists of two main recurrent neural networks (RNNs): an encoder and a decoder. The encoder reads the source sentence and compresses it into a fixed-size context vector, which is then passed to the decoder to generate the target sentence. While effective for short sentences, this fixed-context approach becomes a bottleneck for longer sequences, as it struggles to retain all necessary information.

### 1.2 Related Works

To mitigate the limitations of the fixed-context vector, Bahdanau et al. (2014) introduced the "attention mechanism." This mechanism allows the decoder to selectively focus on different parts of the source sentence at each step of the translation process, creating a dynamic context vector. This innovation dramatically improved the translation of long sentences and has become a standard component in NMT architectures. Subsequently, Luong et al. (2015) proposed other variations of the attention mechanism. The most significant leap in NMT architecture came with the Transformer model (Vaswani et al., 2017), which eschewed RNNs entirely in favor of a self-attention mechanism, setting a new state-of-the-art. This project, however, focuses on the foundational GRU-based Seq2Seq model with Bahdanau attention to demonstrate the core principles of NMT.

## 2 Method

This section outlines the methodology used to build the Korean-to-English translator, from data preparation to the model's architectural details.

## 2.1 Dataset and Preprocessing

The model was trained on a parallel corpus of Korean-English sentences. The preprocessing pipeline involved several key steps:

1. **Cleaning:** Punctuation was standardized, and sentences were cleaned to remove extraneous characters.
2. **Tokenization:** Sentences were tokenized into words. For each language, a vocabulary was built, mapping each unique word to an integer index. Special tokens such as `<start>`, `<end>`, and `<pad>` were added to the vocabulary.
3. **Padding:** Since neural networks require inputs of a fixed size, all sequences in a batch were padded to the length of the longest sequence in that batch using the `<pad>` token.

## 2.2 Model Architecture

The model is an encoder-decoder network implemented using Gated Recurrent Units (GRUs), enhanced with a Bahdanau attention mechanism.

### 2.2.1 Encoder

The encoder processes the input Korean sentence. It is a bidirectional GRU layer that reads the input sequence in both forward and backward directions. This allows the hidden state at each timestep to capture contextual information from both past and future words.

The encoder takes a sequence of token indices $X = (x_1, x_2, \ldots, x_m)$ as input. After passing through an embedding layer, the sequence is fed into the bidirectional GRU. The encoder produces a sequence of hidden states, or outputs, $H_e = (h_1, h_2, \ldots, h_m)$, and final hidden states for both directions.

### 2.2.2 Bahdanau Attention

The Bahdanau attention mechanism computes a context vector, $c_i$, for each decoding step $i$. This context vector is a weighted sum of the encoder's hidden states, where the weights signify the relevance of each source word to the current target word being generated.

The attention score, $e_{ij}$, between the decoder's previous hidden state, $s_{i-1}$, and the encoder's j-th hidden state, $h_j$, is calculated as:

$$e_{ij} = v_a^T \tanh(W_1 h_j + W_2 s_{i-1})$$

where $W_1$, $W_2$, and $v_a$ are trainable weight matrices.

The alignment weights, $\alpha_{ij}$, are then computed by applying a softmax function over these scores:

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{m} \exp(e_{ik})}$$

Finally, the context vector $c_i$ is the weighted sum of the encoder outputs:

$$c_i = \sum_{j=1}^{m} \alpha_{ij} h_j$$

### 2.2.3 Decoder

The decoder is a unidirectional GRU that generates the English translation one word at a time. At each timestep $i$, the decoder receives the previously generated word, its own previous hidden state $s_{i-1}$, and the context vector $c_i$ as input.

The input to the decoder's GRU at step $i$ is the concatenation of the context vector $c_i$ and the embedding of the previous target word $y_{i-1}$. The GRU updates its hidden state $s_i$. The new hidden state is then passed through a fully connected layer with a softmax activation to produce a probability distribution over the entire target vocabulary. The word with the highest probability is chosen as the output for that timestep.

### 2.3 Training

The model was trained end-to-end using the Adam optimizer. The loss function used was Sparse Categorical Crossentropy, which is suitable for multi-class classification problems where the classes are mutually exclusive. The objective is to minimize the difference between the predicted probability distribution and the actual one-hot encoded target word. The training was performed over multiple epochs until the validation loss stabilized.

## 3 Result

The performance of the trained model was evaluated qualitatively by providing it with Korean sentences not seen during training and observing the English translations it produced.

The model demonstrated a clear ability to capture semantic and syntactic structures from the source language and translate them into coherent English. Below are some examples of translations generated by the model.

```
Korean Source:  .
Model Output:  obama is the president .
Expected:      obama is a president.

Korean Source:    .
Model Output:  citizens live in the city .
Expected:      citizens live in a city.

Korean Source:   .
Model Output:  i don t need coffee .
Expected:      i don't need coffee.
```

As shown, the model successfully translates the core meaning of the sentences. The training process was monitored by plotting the loss over epochs, which showed a steady decrease, indicating that the model was learning effectively from the training data. While a quantitative metric like the BLEU score was not calculated, the qualitative results are promising and validate the chosen methodology.

## 4 Conclusion

### 4.1 Discussion

In this project, we successfully built and trained a Korean-to-English translator using a sequence-to-sequence model with a Bahdanau attention mechanism. The results indicate that the model learned to align source and target sentences and can produce meaningful translations. The attention mechanism was crucial, allowing the model to handle dependencies between distant words in the source and target sequences. The quality of the translations is reasonable for a baseline model, though they are often simple and literal.

### 4.2 Future Works

There are several avenues for improving the current model:

1. **Larger Dataset:** Training on a much larger and more diverse parallel corpus would significantly improve the model's fluency and ability to handle a wider range of vocabulary and sentence structures.

2. **Advanced Architecture:** Implementing a more modern architecture like the Transformer, which uses self-attention, could yield substantial performance gains.

3. **Beam Search:** For inference, instead of using a greedy approach (picking the most likely word at each step), implementing beam search would allow the model to explore multiple translation possibilities and choose the one with the highest overall probability.

4. **Quantitative Evaluation:** A more rigorous evaluation using standard metrics like BLEU (Bilingual Evaluation Understudy) score would provide a quantitative measure of translation quality and facilitate comparison with other models.

# 5 Acknowledgment

We acknowledge the AIFFEL program and the author of the original Jupyter Notebook, hjaelee01, for providing the foundational code and learning materials that made this project possible.

# References

[1] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems 27*.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv preprint arXiv:1409.0473*.

[3] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. *arXiv preprint arXiv:1508.04025*.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. In *Advances in Neural Information Processing Systems 30*.