Project: Distant language machine translation

Zilong Wu, Jianing Chen and Xuchen Hu (Group #4)
Electrical and Computer Engineering Department
New York University
Brooklyn, NY 11201
{zw2599, jc10034, xh2090}@nyu.edu

Abstract

This research aims to propose a new architecture that improves the performance for distant language translation. We design a contrastive learning model for Japanese to English translation. The principle idea behind the model is to make the embedding of Japanese sentences similar to the relevant English sentences thus the backbone model would treat the Japanese embedding same as English. We add a multilayer perceptron on the Japanese embedding to mimic the format of English embedding and learn the rule of English through the backbone model. The new designed model shows better performance in Japanese to English translation.

1 Introduction

Translating distant languages is a tough problem for machine translation. Distant language means that the structure of two different languages have huge divergences and the representations of embedding could differ greatly. For example in Japanese to English translation “りんごを食べたい—I want to eat an apple”. The sequence of those two sentences is totally different, the “apple” which should be translated in the front of the sentence dropped at the end. Instead, the subject ‘I’ which is omitted in the Japanese appears at the very start position in the English translation. These kinds of gaps between two languages are obstacles for neural machine translation. How to overcome this problem is researched in this project.

Neural network models have made great achievements in natural language processing, transformer model [1] which was proposed in 2017, utilizes vanilla attention mechanisms in neural networks to fetch features. With the introduction of transformer, more and more powerful language models have been proposed in natural language processing. Shortly after the invention of transformers, Bert [2] moves one step forward. It incorporates the encoder of the transformer to predict the masked vocabulary in the input sentences. Bert achieves shocking performance by leveraging a masked language model over transformer model and advances state of the art performance over eleven natural language processing tasks. However, with those powerful language models, the BLEU score [3] with distant languages is not as high as similar languages such as France and English. In this project, we propose a contrastive learning based model to improve the performance of distant language translation.

Contrastive learning is a popular form of self-supervised learning that encourages augmentations of the same input to have more similar representations compared to augmentations of different inputs. Contrastive learning aims to learn effective representations by pulling semantically close neighbors together and pushing the non-related neighbors away from each other. This architecture was first used in computer vision for visual representation, encouraged by the success of contrastive learning in computer vision, researchers in natural language processing started to apply this structure in text classification and other tasks in natural language processing. SimCSE [4], a simple contrastive learning of semantic embedding which predicts input sentences itself with dropout used as noise. By inputting the sentences into dropout pretrained encoder twice, SimCSE acquires two different embeddings as positive pairs. Then other sentences from the same mini-batch are treated as negatives

and the language predicts the positive pairs among the negative sentences. This simple architecture outperforms the next sentence. Inspired by SimCSE, this project constructs a contrastive architecture for machine translation.
In this project, we make two main contributions to distant language translation.
(1) We incorporate the contrastive learning in machine translation. The source language sentence and the corresponding target language sentence are treated as augmentations of same input. The objective of contrastive learning is to make the embedding of the input pair have similar representations compared to different source sentences.
(2) We utilize a Multilayer perceptron (MLP) to mimic the source language sentences to the target language sentences. In this case, we only need to train paramters in MLP.
(3) We use the multilingual model to put the translated English sentences into the transformer model and translate itself into English which increases the performance of translation.

2 Related work

In this section, we would generally review some approaches related to machine translation and contrastive learning.

2.1 Tokenization

Tokenization is an indispensable step in natural language processing (NLP) tasks. Tokenizing English or word based language is not difficult, people only need to segment different words by dividing the sentences with space. While for other languages like Chinese and Japanese, there is no space between two individual words. In such circumstances, how to tokenize the special language is significant in translation. In this project, we utilize the fugashi a Japanese tokenize tool to divide words in Japanese sentences.

2.2 Sequence to Sequence learning

In machine translation both the input and output are a variable-length sequence. To address this type of problem, Sequence to Sequence learning (Seq2Seq) [5] is proposed to convert sequence from one domain to another. The encoder-decoder architecture allows the input and output have different length sequence.

2.3 Transformer model

Transformer [1] is a popular language model for natural language processing tasks. The main idea of Transformer model is to eliminate the long term dependency in previous neural networks such as long short term memory (LSTM) [6]. Transformer model utilizes only attention mechanism to build the neural networks and allows the model to learn all knowledge from all positions of sentences. In machine translation, transformer incorporates the Seq2Seq learning and build the encoder-decoder structure. The difference between the transformer mode and the previous Seq2Seq model is that transformer uses the attention mechanism while the former model uses recurrent neural networks.

2.4 Contrastive learning

Contrastive learning is a form of self-supervised learning that encourages augmentations of the same input to have more similar representations compared to augmentations of different inputs. The purpose of contrastive learning is to distinguish the features of different inputs, that the original input and augmented input would learn to have higher score than different input with the augmented input. In this project, the original input and the augmentation becomes the source language and the target language. The the pair of source sentence and target sentence would have higher similarity while the different source sentences with the given target sentence would have lower similarities.

2.5 dataset

In this project we use a Business Scene Dialogue (BSD) [7] dataset for translation, a Japanese-English parallel corpus containing written conversations in various business scenarios. The dataset

was constructed in 3 steps: 1) selecting business scenes, 2) writing monolingual conversation scenarios according to the selected scenes, and 3) translating the scenarios into the other language. Half of the monolingual scenarios were written in Japanese and the other half were written in English.

3 Model architecture

In this section, we would explain the whole architecture of our model and the main idea behind this model.
The neural machine translation process could be viewed as a simple function $y = F(x)$, where y is the output and x is the input word embedding from a different language. F represents the language model which catches semantic information from source language and translates it to the results.Suppose the input language is the same as the output language, x is y' s embedding vector and the output of the language model should perfectly match the target sentence. Thus if there is a function g that makes $g(x)$ approximately equal to y' s embedding, the performance of the language model should be improved. This proposal builds a contrastive learning architecture to fulfill the approximation concept.

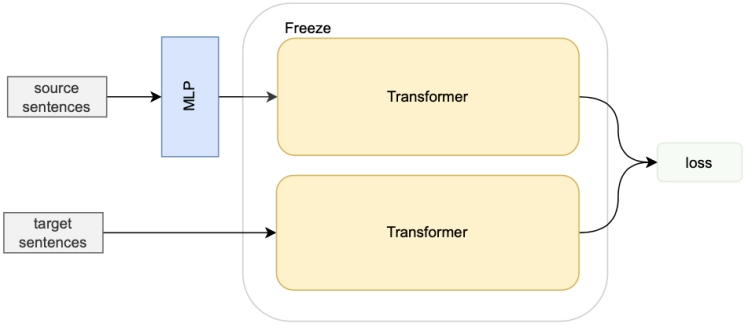

Figure 1: Architecture of contrastive learning for contrastive learning

Figure 1 shows the architecture of contrastive learning, source language and target language are inputted into the same freezed transformer encoder. The difference between them is that the source sentences are modified by a multilayer perceptron (MLP) layer which reshape the source embeddings to 'target embeddings'. The reason why we add a MLP is that we want to mimic the vector of target sentences based on the source sentences. By training the parameters within MLP layers, the MLP would learn how to transform the source vector to target vector. The position of MLP layer could be placed at the input of target encoder, however, since we want to transform the source embedding to target, the MLP layer would not be placed on target side.
In training process, we construct the transformer model into multilingual architecture which means that the transformer could absorb both English sentences and Japanese sentences. To achieve this architecture, we enlarge the source vocabulary size thus all words from two languages could be transformed into vectors.
Figure 2 shows the entire translation process. In contrastive learning, we only train the parameters in MLP layers, the other parameters in transformer model do not change. In the translation process, the MLP would keep the parameters learned from contrastive learning and transform the original encoder embedding to 'target' encoder embedding. During the whole translation process, no paramters would be updated since we only mimic the source language to the target language.

4 Experiment

4.1 Changes in final project

At the very beginning of the project, we decide to put the multilayer perceptron (MLP) layer in front of the transformer model. Figure 3 shows the initial attempt, the MLP layer would transform the word embedding of source sentences to the 'target embedding'. However, in the later learning

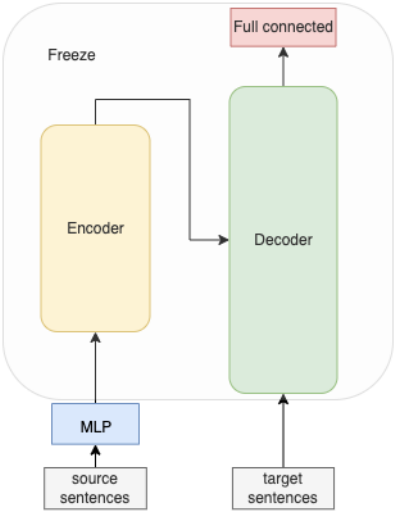

Figure 2: Architecture of transformer based translation model

of machine translation and transformers, the initial embedding commonly do not use word2vec [8] or Glove [9]. The word vectors generated by those embedding methods are hard to use in machine translation. Thus we changed the architecture of our model in the midterm report, we placed the MLP layer in the middle of the transformer as shown in Figure 3, within the encoder part and the decoder part. However, in the experiment the result of this architecture behaves worse than expected. The embedding of input sentences become unstable and the translated sentences have bad result which repeat same words in a line or have zero output as result. The bleu scores are approximately zero. We need to reconsider about the whole structure of the model. Then we consider about the multilingual model of translation which can embedding both the source and target languages. By constructing a multilingual model, we could transform the Japanese embedding similar to the English embedding. Since they could be represented by the same transformer encoder. The key idea of constructing this structure is to separate the embedding of words out of the main transformer model.

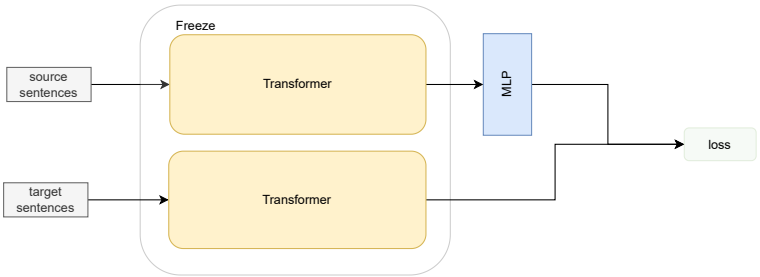

Figure 3: Architecture of initial attempt

Besides the structure modifications, in the midterm report we change the dataset of this project. We tend to use the OPUS open corpus dataset at first. Unfortunately, the OPUS only has English to Japanese dataset. We change the dataset to Business Scene Dialogue which translate from Japanese to English. The direction of source language and target language is important, since we need to compare the ability of translation from the native language to another language. The Business Scene Dialogue consists of 20000 training Japanese English pairs and 2000 test pairs. The Japanese vocabulary has over 5000 words and the relevant English words includes over 6000 words and the multilingual model consists over 12000 tokens.

4.2 Transformer

In this section we would explain the detail parameters and architecture of our transformer model. Due to the limited calculation resources, we build a simple transformer model which smaller than the base transformer. Our own model only includes two blocks for both the encoder and decoder composition. The key value and query size and the hidden size of the word embedding are 32 and the dropout value is 0.1. The number of steps for each sentences are 10 which means at most 10 words would be counted in a sentence.
The embedded sentences would be input into the encoder of our translation model and calculate the attention for each words. After the attention mechanism, to protect the stable structure of the model layer normalization is needed. The normalized vectors would be streamed into a residual multi layer perceptron layer. The output of this block would be input into the next block.
The decoder part of transformer is almost identical to the encoder part, the difference between the encoder and decoder part is that the input of the decoder part is the output of the encoder. And in the training process, the manual translated sentences would also the input of the decoder. While in testing procedure, the manual translated sentences would be replaced by the begin of the sentence token.

4.3 Contrastive MLP structure

The contrastive structure is just a simple MLP layer which modifies the input embedding towards to the 'target' language embedding. The MLP layer would be placed after the embedding layer of encoder. The MLP layer would transform the pretrained embedding of source sentences and output the transformed embedding into the encoder of transformer. The loss function to train this structure is same with the translation function while the label would be replaced by the English embedding and translated English words.

5 Results

5.1 Clean dataset

We extract the Japanese sentences and correlated English sentences out of the raw datasets and tokenize them through different tokenization method. For Japanese vocabulary, we use the fugashi [10] to process the Japanese sentences. For English sentences, we just lower the case of words and split them with space. The over all tokens lengths are shown in figure 4. Since most of sentences compose less than 20 words, in the experiment procedure we would chose 10 as number of steps. Figure 5

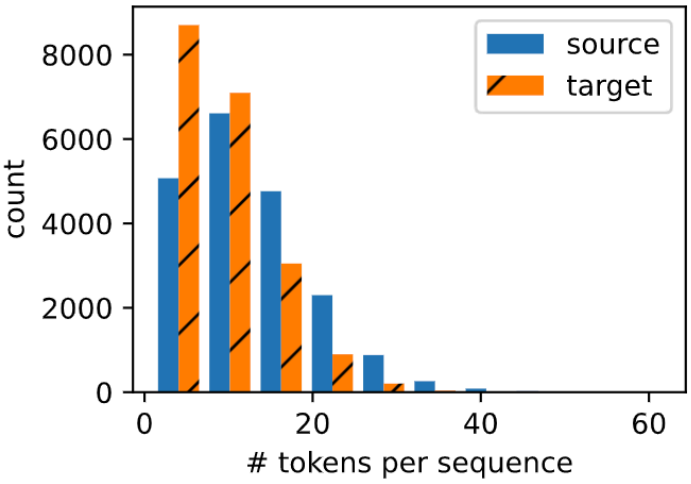

Figure 4: Length for Japanese sentences and English sentences

```
([['はい', '、', 'k', '社', 'システム', '開発', '部', 'です', '。'],
  ['h', '社', 'の', '高市', 'と', '申し', 'ます', '。'],
  ['いつ', 'も', 'お', '世話', 'に', 'なっ', 'て', 'おり', 'ます', '。'],
  ['こちら', 'こそ', '、', 'お', '世話', 'に', 'なっ', 'て', 'おり', 'ます', '。'],
  ['稲田', 'さん', 'は', 'いらっしゃい', 'ます', 'か', '？']],
 [['hi',
   'this',
   'is',
   'the',
   'systems',
   'development',
   'department',
   'of',
   'company',
   'k.'],
  ['my', 'name', 'is', 'takaichi', 'from', 'company', 'h.'],
  ['thank', 'you', 'as', 'always.'],
  ['thank', 'you', 'as', 'always', 'as', 'well.'],
  ['is', 'inada-san', 'there?']])
```

Figure 5: Tokenization of sentences

shows the tokenization results for both Japanese sentences and English sentences. By build a vocabulary library for source and target languages we could transform the word tokens into vectors which record each words in sentences.

5.2 Translation result

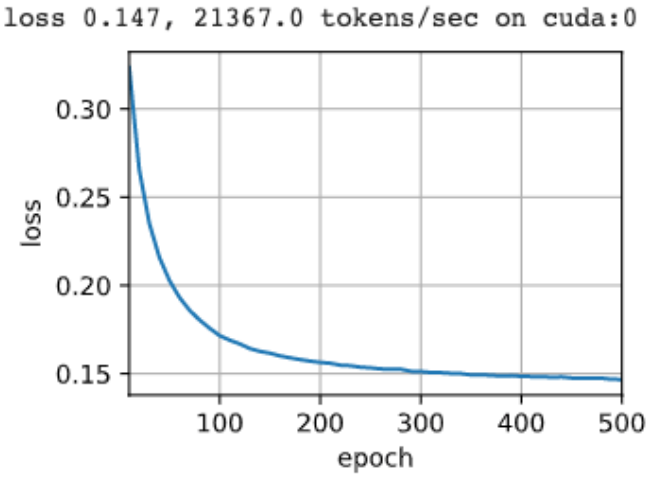

Figure 6: Loss of the simple transformer model

Figure 6 shows the loss result of the Transformer model, bleu score of the Japanese translation is 7.23 and the English to English translation bleu score is 35.34. While in the pure Japanese to English translation the score is 0.84 for 200 epoches. Even though the score of translation is not good as expected due to the simple architecture, the improvement of translation is apparent in comparing with the monolingual translation. In the figure 7 we could see the effect of the translation model. The translated sentences could partially interpret the meaning of the Japanese language. However, some long sentences or complicated sentences it could not generate accurate meaning.

はい、K社システム開発部です。 , yes, this is <unk> from k university.
H社の高市と申します。 , my name is takaichi from company h.
いつもお世話になっております。 , thank you for your support.
こちらこそ、お世話になっております。 , thank you for your name is nose from you?
稲田さんはいらっしゃいますか? , is <unk> there?
1時間ほど前に、お電話いただいたそうなんですけど。 , i will call you a call around 1 hour.
はい、代わります。 , yes, please check the end of country.
少々お待ちください。 , please wait a moment.
稲田さん、H社の高市様からお電話です。 , hello, you've reached the call from h <unk> from h
もしもし、稲田です。 , hello, this is <unk>
H社の高市です。 , h is takaichi from company h.

Figure 7: Translation

5.3 model performance

	contrastive model	monolingual model	mutiligual model
bleu score ja-en	7.23	0.84	7.56
bleu score en-en	35.34	44.45	34.45

Table 1: Performance of different translation models

In this section, we would compare the performance of our model and the monolingual models. Table 1 shows the performance of translation for contrastive learning model, monolingual model and the multilingual model. We could see that multilingual model and the contrastive learning model behaves way better in Japanese to English translation. While the English to English translation would be less effective than the monolingual model. That's the reason Japanese is a distance lanugage and it is difficult to translate in a simple transformer, however the English to English would be perfect in this case. The transformer would perform good when calculate $y = y$

6 Conclusion

In this project, we design a contrastive learning based multilingual model to translate a distance language – Japanese to English. The performance of transformer model behaves way better than the simple monolingual model and we could understand partially the meaning of original sentences in Japanese. However, we still have a long way to go in distant language translation, we could enlarge the scale of the model or utilize the other advanced techniques to refine the model such as prompt. It's a tough experience for build a contranstive learning translation model for distance languages, we learned a lot from this project and we hope we could do it better in the near future.

References

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[3] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.

[4] Tianyu Gao, Xingcheng Yao, and Danqi Chen. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*, 2021.

[5] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

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

[7] Matīss Rikters, Ryokan Ri, Tong Li, and Toshiaki Nakazawa. Designing the business conversation corpus. In *Proceedings of the 6th Workshop on Asian Translation*, pages 54–61, Hong Kong, China, November 2019. Association for Computational Linguistics.

[8] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[9] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[10] Paul McCann. fugashi, a tool for tokenizing japanese in python. *arXiv preprint arXiv:2010.06858*, 2020.