Neural Machine Translation

Gaurav Rajesh Parikh

March 28, 2024

1 Tokenization

I used Byte Pair Encoding to explore a range of vocabulary sizes. Based on Li et al. [2020], BPE expands rare words into two or more sub words thereby lengthening a sequence, and merges frequent-character sequences into one sub word piece thereby shortening a sequence. As suggested in the paper, NMT can be viewed as a form of structured classification (with the target language being class labels) and the merging or splitting leads to greater class balance for classification. I expected that if the train data had a lot of rare words, increasing vocabulary size should help as it would capture more granularity across these words, while if we have a lot of common words shrinking the vocabulary should be a better approach to achieving class balance by allowing more common words to be merged. I observed a lot of very common words with high frequency so I decided to explore shrinking the vocabulary to different levels. I expected to see that a smaller vocabulary should improve balance and I found that this was indeed true as I achieved a BLEU score of **39.61738** on the validation set using this set up for a vocabulary size of 1000.

1.1 Vocabulary Size(`vocab_size`)

I further tried to change the vocabulary size to 1000, 2000, 4000, 8000 and 16000 while using Byte Pair Encoding to explore the effect of vocab size selection on the BLEU Score. The results are summarized in the following table.

Vocabulary Size	BLEU Score
1000	**39.61738**
2000	38.27882
4000	36.6637
8000	35.42821
16000	35.6628

Table 1: Vocabulary Size and BLEU Score

2 Hyperparameters

2.1 Effect of Hidden Dimension (`hidden_dim`)

I experimented with three values for `hidden_dim`: 128, 256 (baseline), and 512. I expected that increasing the hidden dimension might improve translation quality marginally since it allows for a richer learned representation in the latent space where the input sequences are encoded but I did not see this in my results as I obtained a BLEU score of 36.72822 for 512 validation with attention. I presume that this is because the dataset is fairly small and increasing dimension is causing overfitting.For a lower dimension of 128, the score decreases slightly but not significantly to 34.167171 presumably as the learned representation is not as rich.

2.2 Influence of Dropout Rate (`dropout`)

Dropout regularization is crucial for preventing overfitting in neural networks as it drops a fraction of values in the linear layer. I experimented with dropout rates of 0.2, 0.3 (baseline), and 0.4. I expected that if the model was overfitting, increasing the dropout rate to 0.4 should yield better results. However, I found that by decreasing to 0.2 I was able to get slightly better performance which implies that a slightly lower dropout rate can help the model generalize better and the model might have been under fitting. I got BLEU scores for 37.29283 35.128681, and 34.22893 respectively for dropouts 0.2, 0.3 and 0.4.

3 Final Model

My final model is a beam search model with attention with BPE with a vocab size of 1000 with dropout 0.2 that gives a **test BLEU score of 39.055**. I tried other combinations of approaches but settled on these parameters as they gave best results I could obtain within my compute constraints.

References

Chengcheng Li, Yu Wu, Han Wu, Hui Li, and Qun Liu. Improving the efficiency and effectiveness of human fact verification with deep learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020. URL `https://aclanthology.org/2020.findings-emnlp.352.pdf`.