Machine Translation Model Comparison

Shreshth Vishwakarma
Roll Number: 24095105
CSOC IG

June 21, 2025

1 Model Descriptions

Part 1: Encoder-Decoder without Attention

Basic sequence-to-sequence architecture using LSTM layers. I used single layered LSTM.

Part 2: Encoder-Decoder with Luong Attention

Added attention mechanism to dynamically align decoder output with relevant encoder states.
Teacher forcing ratio=0.5 (half the time feed ground-truth token)
Loss: CrossEntropyLoss
Optimizer: Adam
Mixed precision: torch.cuda.amp.autocast() + GradScaler
Batch size: 8 (dynamic padding for minimal wasted tokens)

2 Special Notes

I could not train the model completely hence was not able to improve the model nor was i able to perform the bonus task. I wrote the code but was not able to train the model. Also the transformer model task was not done. All of this was not done due to time limitations due to prior commitments and persistent training problems with google colab due to its restricted RAM and usage time limitations.

Improvement Strategies

- I did not use bidirectional LSTMs (which was used in the original paper by Ilya Sutskever) in encoder.
- Also the number of epochs were less, which could have been increased for better training.

- More training data could be used, i used only ——- number of sequences (which is very less for the model to learn) as it was taking lot of time.
- I used single layer LSTM, but originally 4 layers were stacked, so similar architecture could have been followed for better performance.
- These methods and strategies could have been easily applied and it should have definitely improved the performance of the model. ALL of this was not done due to time limitations, due to prior engagements(i should have managed my time in a better way).