# Assignment 4 Writeup

**DO NOT TAG**

Name:
GT Email:

# Seq2Seq Results – Default configuration

## RNN

Training Loss: 4.2847
Training Perplexity: 72.5790
Validation Loss: 4.3471
Validation Perplexity: 77.2522

## LSTM

Training Loss: 3.0410
Training Perplexity: 20.9267
Validation Loss: 3.1956
Validation Perplexity: 24.4254

## RNN-with-Attention

Training Loss: 3.5171
Training Perplexity: 33.6873
Validation Loss: 3.5827
Validation Perplexity: 35.9700

## LSTM-with-Attention

Training Loss: 2.9781
Training Perplexity: 19.6496
Validation Loss: 3.1269
Validation Perplexity: 22.8035

# Seq2Seq Explanation (RNN vs LSTM)

Compare your RNN result to your LSTM result and explain why they differ.

LSTM has both lower perplexity and loss than RNN. The reasons are as below:

LSTM is designed to address the vanishing gradient problem which is a common issue in traditional RNNs. It can capture long-term dependencies via memory cells, which can store information over long periods without degradation. The LSTMs' gates regulate the information flow by selectively allowing certain information to pass through and inhibiting others. This ability to selectively update and access the memory cells helps LSTMs better retain important information over long sequences. Also, LSTMs are generally easier to train compared to traditional RNNs due to their improved ability to handle long-range dependencies.

# Seq2Seq Explanation (RNN vs RNN-with-Attention)

Compare your RNN result to your RNN-with-Attention result and explain why they differ.

RNN-with-Attention has both lower perplexity and loss than RNN. The reasons are as below:
Attention mechanisms allow the model to selectively focus on relevant parts of the input sequence while generating an output at each time step. This selective focus enables the model to effectively capture long-range dependencies and attend to important information. Additionally, by allowing the model to attend to specific parts of the input sequence at each time step, attention mechanisms effectively reduced context fragmentation.

# Seq2Seq Results – Best model

Your best model after hyper-parameter tuning

Best model

Training Loss: 2.6863

Training Perplexity: 14.6778

Validation Loss: 3.1242

Validation Perplexity: 22.7416

List your best model hyper-parameter values including model type:

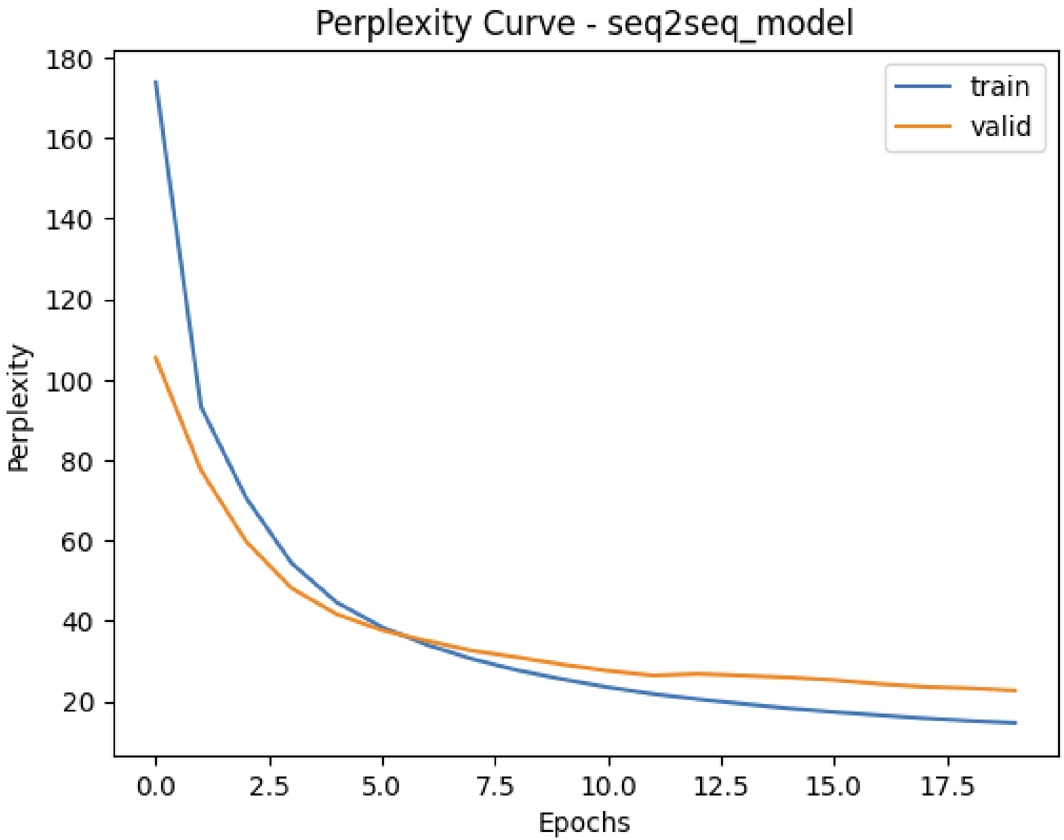
encoder_emb_size = 256, encoder_hidden_size = 256, encoder_dropout = 0.3,

decoder_emb_size = 256, decoder_hidden_size = 256, decoder_dropout = 0.3,

learning_rate = 1e-3, model_type = "LSTM“, attention=True,

EPOCHS = 20

# Seq2Seq Best model Learning Curves (Perplexity)



# Seq2Seq Explanation – Best model

Explain the details of your best model. Explain what you did to improve your model's performance and why

I used the LSTM model with attention, which has the best performance for now with the default set up among the four models. I increased the hidden dimension of both encoder and decoder to allow the model to capture more complex patterns in the data. I also increased embedding size to allow for more expressive power in the embeddings. Because the validation loss and perplexity have already been almost the same as the training loss and perplexity when the hidden dimension is set to 128. Now I raised the hidden dimension to 256. Therefore, to prevent overfitting, I increased the dropout rate to 0.3. I kept the batch size and epochs as default.

# Transformer Results

## Default configuration (Encoder Only)

Training Loss: 2.1267

Training Perplexity: 8.3872

Validation Loss: 2.9561

Validation Perplexity: 19.2228

## Default configuration (full transformer)

Training Loss: 1.3247

Training Perplexity: 3.7609

Validation Loss: 1.5983

Validation Perplexity: 4.9447

## Best model (full transformer)

Training Loss: 0.9865

Training Perplexity: 2.6817

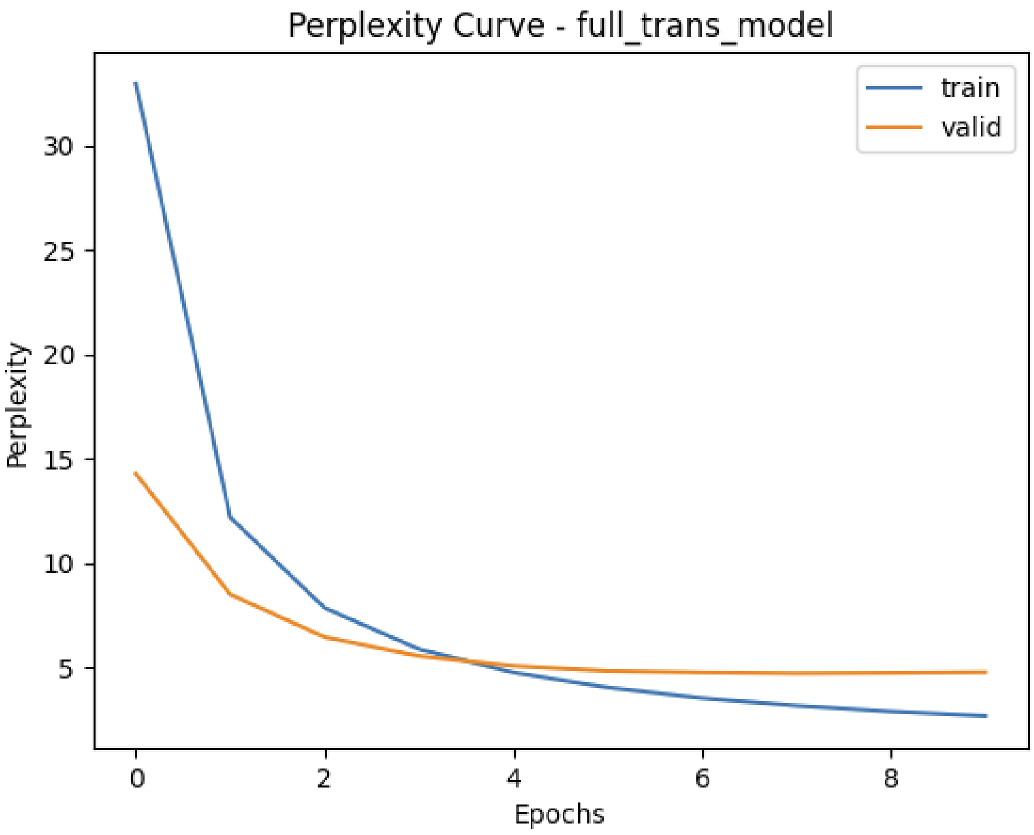
Validation Loss: 1.5620

Validation Perplexity: 4.7682

## List your best model hyper-parameter values (full transformer):

hidden_dim=256, num_heads=4, dim_feedforward=2048, num_layers_enc=2, num_layers_dec=2, dropout=0.2, max_length=43, ignore_index=1, learning_rate = 1e-3, EPOCHS = 10

# Full Transformer Best model Learning Curves (Perplexity)



# Full Transformer Explanation – Best model

Explain what you did here and why you did it to improve your model performance.

I increased hidden dimension to increase the capacity of the model, because it allows the model to represent more complex patterns and relationships in the data. With a larger hidden dimension, the model can learn richer feature representations, which might lead to better generalization and performance on both the training and validation sets. I also increased the number of attention heads allows the model to attend to more diverse aspects of the input sequence simultaneously. Because with more attention heads, the model can capture more fine-grained and multi-faceted relationships within the data. I also tried to increase the dropout rate to prevent potential overfitting but found the model's performance became worse. Therefore, I kept the dropout rate as default.

# Transformer (Encoder Only)  Translation Results

Put translation results for your model (1st 9 sentences) here

| Input sentence | Back translation |
|---|---|
| ['<sos>', 'a', 'man', 'in', 'an', 'orange', 'hat', 'starring', 'at', 'something', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'man', 'in', 'an', 'orange', 'hat', 'an', 'something', 'something', '\n', '.', '\n', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>'] |
| ['<sos>', 'a', 'boston', 'terrier', 'is', 'running', 'on', 'lush', 'green', 'grass', 'in', 'front', 'of', 'a', 'white', 'fence', '.', '\n', '<eos>', '<pad>'] | ['<sos>', 'a', 'boston', 'of', 'runs', 'in', 'the', 'grass', 'in', 'in', 'front', 'white', 'white', 'fence', '.', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>']' |
| ['<sos>', 'a', 'girl', 'in', 'karate', 'uniform', 'breaking', 'a', 'stick', 'with', 'a', 'front', 'kick', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'girl', 'in', 'a', 'a', 'red', 'a', 'a', 'with', 'a', 'a', '.', '\n', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>'] |
| ['<sos>', 'people', 'are', 'fixing', 'the', 'roof', 'of', 'a', 'house', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'people', 'are', 'the', 'roof', 'a', 'of', 'a', '.', '\n', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>'] |
| ['<sos>', 'a', 'group', 'of', 'people', 'standing', 'in', 'front', 'of', 'an', 'igloo', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'group', 'of', 'people', 'standing', 'in', 'front', 'front', 'of', '.', '\n', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>'] |
| ['<sos>', 'a', 'guy', 'works', 'on', 'a', 'building', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'guy', 'works', 'on', 'a', 'building', 'building', '\n', '\n', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>'] |
| ['<sos>', 'a', 'man', 'in', 'a', 'vest', 'is', 'sitting', 'in', 'a', 'chair', 'and', 'holding', 'magazines', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'man', 'in', 'a', 'vest', 'a', 'sitting', 'sitting', 'chair', 'chair', 'and', 'chair', '.', '.', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>'] |
| ['<sos>', 'a', 'mother', 'and', 'her', 'young', 'song', 'enjoying', 'a', 'beautiful', 'day', 'outside', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'mother', 'and', 'her', 'blond', 'son', 'enjoying', 'outdoors', 'beautiful', 'sunny', 'day', 'sunny', '\n', '<eos>', '\n', '<eos>', '<eos>', '<eos>', '<eos>'] |
| ['<sos>', 'a', 'woman', 'holding', 'a', 'bowl', 'of', 'food', 'in', 'a', 'kitchen', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'woman', 'wearing', 'a', 'kitchen', 'a', 'kitchen', 'kitchen', 'kitchen', 'food', '.', 'kitchen', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>'] |

# Full Transformer Translation Results

Put translation results for your best model (1st 9 sentences) here

| Input sentence | Back translation |
|---|---|
| ['<sos>', 'a', 'man', 'in', 'an', 'orange', 'hat', 'starring', 'at', 'something', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'man', 'with', 'an', 'an', 'orange', 'orange', 'hat', 'hat', '.', '.', '\n', '\n', '<eos>', '<\n>'] |
| ['<sos>', 'a', 'boston', 'terrier', 'is', 'running', 'on', 'lush', 'green', 'grass', 'in', 'front', 'of', 'a', 'white', 'fence', '.', '\n', '<eos>', '<pad>'] | ['<sos>', 'a', 'boston', 'gray', 'with’, 'dog’, 'two’, 'running', '<unk>','in’, 'in', 'front', 'front’, 'of’, 'of’, 'a’, 'grass’, 'white','\n'], |
| ['<sos>', 'a', 'girl', 'in', 'karate', 'uniform', 'breaking', 'a', 'stick', 'with', 'a', 'front', 'kick', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', ‘girl', 'with', 'uniform', 'breaks', 'stick', 'by’, 'by’, 'use', 'use', 'of', 'foot', '.', '\n', '<eos>'] |
| ['<sos>', 'people', 'are', 'fixing', 'the', 'roof', 'of', 'a', 'house', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'many', 'person', 'on', 'roof', 'from', 'from', 'a', 'a', 'building', 'is', '.', '\n', '<eos>'] |
| ['<sos>', 'a', 'group', 'of', 'people', 'standing', 'in', 'front', 'of', 'an', 'igloo', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'group', 'of', 'person', 'stand', 'outside', '<unk>', 'a', 'a', 'snow', '.', '.', '\n’, '<eos>'] |
| ['<sos>', 'a', 'guy', 'works', 'on', 'a', 'building', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'worker', 'works', 'on', 'on', 'the', 'the', 'structure', '.', '\n', '<eos>', '<\n>'] |
| ['<sos>', 'a', 'man', 'in', 'a', 'vest', 'is', 'sitting', 'in', 'a', 'chair', 'and', 'holding', 'magazines', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'person', 'in', 'a', 'vest', 'is', 'sitting', 'on’, 'on', 'a', 'a', 'chair', '.’, '.', '\n', '<eos>’] |
| ['<sos>', 'a', 'mother', 'and', 'her', 'young', 'song', 'enjoying', 'a', 'beautiful', 'day', 'outside', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'mother', 'and’, 'and', 'her', 'her', 'young', 'song', 'enjoying', 'a', 'beautiful', 'day', 'outside', '.', '\n', '<eos>’] |
| ['<sos>', 'a', 'woman', 'holding', 'a', 'bowl', 'of', 'food', 'in', 'a', 'kitchen', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'woman', 'holding', 'a', 'a', 'bowl', 'bowl', 'of', 'of', 'food', 'food', '.’, '.', '\n', '\n', '<eos>’] |

# Seq2Seq (Best model) Translation Results

Put translation results for your best model (1st 9 sentences) here

| Input sentence | Back translation |
|---|---|
| ['<sos>', 'a', 'man', 'in', 'an', 'orange', 'hat', 'staring', 'at', 'something', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'man', 'wearing', 'a', 'orange', 'hat', 'hat', 'is', 'something', 'something', 'something', '.', '\n', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>'] |
| ['<sos>', 'a', 'boston', 'terrier', 'is', 'running', 'on', 'lush', 'green', 'grass', 'in', 'front', 'of', 'a', 'white', 'fence', '.', '\n', '<eos>', '<pad>'] | ['<sos>', 'a', 'white', 'white', 'grass', 'grass', 'grass', 'grass', 'grass', 'grass', 'grass', 'grass', 'grass', 'grass', 'grass', 'grass', 'grass', 'grass', 'grass', 'grass'] |
| ['<sos>', 'a', 'girl', 'in', 'karate', 'uniform', 'breaking', 'a', 'stick', 'with', 'a', 'front', 'kick', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'girl', 'in', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a'] |
| ['<sos>', 'people', 'are', 'fixing', 'the', 'roof', 'of', 'a', 'house', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'people', 'are', 'the', 'the', 'the', 'the', '.', '\n', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>'] |
| ['<sos>', 'a', 'group', 'of', 'people', 'standing', 'in', 'front', 'of', 'an', 'igloo', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'group', 'of', 'people', 'standing', 'in', 'front', 'of', 'people', 'standing', 'in', 'front', 'of', 'people', 'standing', 'in', 'front', 'of', 'people'] |
| ['<sos>', 'a', 'guy', 'works', 'on', 'a', 'building', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'guy', 'working', 'a', 'building', 'building', 'building', '\n', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>'] |
| ['<sos>', 'a', 'man', 'in', 'a', 'vest', 'is', 'sitting', 'in', 'a', 'chair', 'and', 'holding', 'magazines', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'man', 'in', 'a', 'sitting', 'sitting', 'sitting', 'sitting', 'sitting', 'chair', 'sitting', '.', '.', '\n', '<eos>', '<eos>', '<eos>', '<eos>', '<eos>'] |
| ['<sos>', 'a', 'mother', 'and', 'her', 'young', 'song', 'enjoying', 'a', 'beautiful', 'day', 'outside', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'mom', 'and', 'day', 'day', 'day', 'day', 'day', 'day', 'day', 'day', 'day', 'day', 'day', 'day', 'day', 'day', 'day', 'day'] |
| ['<sos>', 'a', 'woman', 'holding', 'a', 'bowl', 'of', 'food', 'in', 'a', 'kitchen', '.', '\n', '<eos>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>', '<pad>'] | ['<sos>', 'a', 'woman', 'is', 'a', 'kitchen', 'kitchen', 'kitchen', 'kitchen', 'kitchen', 'kitchen', 'kitchen', 'kitchen', 'kitchen', 'kitchen', 'kitchen', 'kitchen', 'kitchen', 'kitchen', 'kitchen'] |

# Compare Transformer (Encoder Only) to Transformer (Full transformer)

Compare your results for default settings for Encoder Only Transformer vs Full transformer both quantitatively and qualitatively. Explain why you see differences.

In terms of quantitative evaluation, the full transformer should have better BLEU scores. In terms of qualitative evaluation, the full transformer should have better fluency, coherence, and semantic accuracy. The encoder-only transformer only generates a list of words without forming a coherent sentence. Here our task is machine translation, where decoding is required, so the full transformer outperforms the encoder-only transformer. The reason is the full transformer can capture bidirectional dependencies and generate output sequences by decoding context vectors produced by the encoder. However, encoder-only transformer may perform better on tasks where only encoding is required, as it does not incur the additional complexity and training requirements of the decoder.

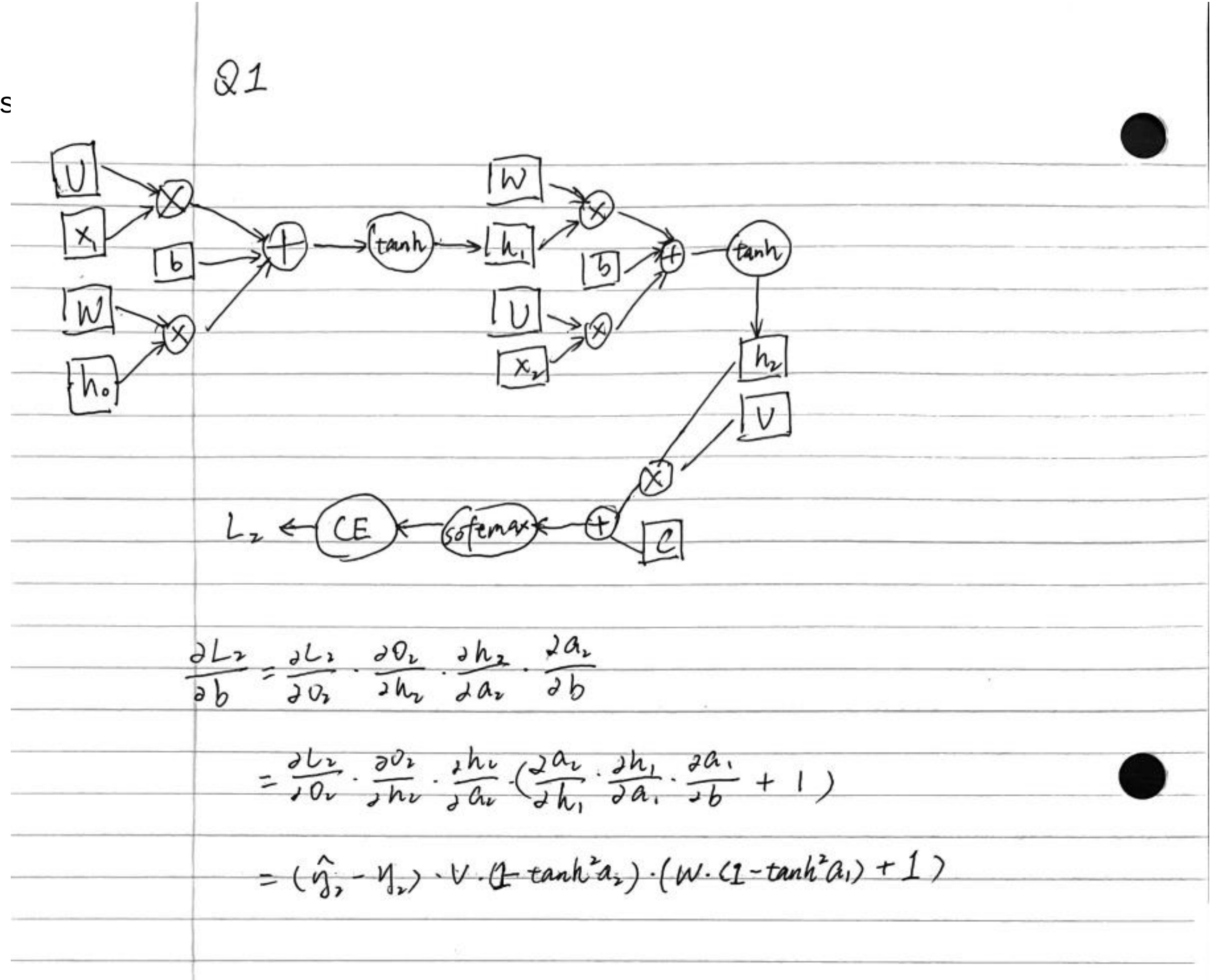
# Compare Seq2Seq to Transformer (Best models)

Compare your Seq2Seq best model results to your Transformer best model results both quantitatively and qualitatively and explain the differences.

In terms of quantitative evaluation, the full transformer has better BLEU scores. In terms of qualitative evaluation, the full transformer has better fluency, coherence, and semantic accuracy. The Seq2Seq model tends to perform well at the first few words of each sentences but its performance becomes worse when sentence becomes longer. As the Seq2Seq model may struggle with accurately translating longer sentences or capturing subtle nuances in the input text due to limitations in modeling long-range dependencies. Full transformer model can effectively capture long-range dependencies and understand the context of the entire input sequence, leading to more accurate and contextually relevant translations.

# Theory question

Add additional slides if necess



Q1

U
x1
×
b
W
h0
×
+
tanh
W
h1
×
b
U
x2
×
+
tanh
h2
V
×
C
+
softmax
CE
L2

$$\frac{\partial L_2}{\partial b} = \frac{\partial L_2}{\partial O_2}\cdot\frac{\partial O_2}{\partial h_2}\cdot\frac{\partial h_2}{\partial a_2}\cdot\frac{\partial a_2}{\partial b}$$

$$= \frac{\partial L_2}{\partial O_2}\cdot\frac{\partial O_2}{\partial h_2}\cdot\frac{\partial h_2}{\partial a_2}\cdot\left(\frac{\partial a_2}{\partial h_1}\cdot\frac{\partial h_1}{\partial a_1}\cdot\frac{\partial a_1}{\partial b}+1\right)$$

$$= (\hat{y}_2 - y_2)\cdot V\cdot(1-\tanh^2 a_2)\cdot(W\cdot(1-\tanh^2 a_1)+1)$$

# Paper discussion

I read the paper “Do Vision Transformers See Like Convolutional Neural Networks?” by Maithra Raghu et al. The review summary is as below:

The main contribution of this paper is to analyze the differences between Vision Transformers (ViTs) and Convolutional Neural Networks (CNNs) in terms of their internal representation structure and their performance on image classification tasks. The research indicates that ViTs possess more consistent representations across all layers compared to CNNs. ViTs incorporate more global information than CNNs at lower layers, which leads to quantitatively different features. Skip connections in ViTs have a stronger influence on performance and representation similarity compared to skip connections within CNNs, such as ResNets. ViTs successfully preserve input spatial information, which is important for tasks like object detection. The paper also provides empirical evidence and experimental results to support its findings. My personal takeaway from this paper is that ViTs and CNNs have distinct characteristics in terms of their internal representation structure and feature learning. ViTs leverage self-attention mechanisms to aggregate global information and exhibit more uniform representations across layers. One weakness of the paper is that it mainly focuses on image classification tasks and does not extensively cover other computer vision tasks.

# Compare and contrast the learned features of ViTs and CNNs? For differences between the two, please provide explanations in terms of network architecture and training.

In terms of the network architecture, ViTs use self-attention mechanisms to aggregate information across locations, allowing them to capture global dependencies in the input data. This enables ViTs to incorporate more global information at lower layers compared to CNNs. In contrast, CNNs rely on convolutional layers that encode spatial equivariance as an inductive bias. This bias enables CNNs to learn local and translation-invariant features.

In terms of the training, ViTs capture long-range dependencies through self-attention, which allows them to model global relationships between image patches. On the other hand, CNNs use local receptive fields in convolutional layers to capture local patterns and spatial hierarchies.

# What is meant by spatial localization? And why might we consider the use of ViTs better for object detection?

Spatial localization refers to the ability of a model to accurately identify and localize objects within an image. ViTs have shown promise in this regard because they successfully preserve input spatial information throughout their layers. This is due to the self-attention mechanism, which allows ViTs to capture long-range dependencies and explicitly model pixel-to-pixel relationships.

The use of ViTs for object detection can be advantageous because their attention mechanisms enable them to capture global context and long-range dependencies, which are important for accurately localizing objects. Traditional CNN-based object detection methods often rely on anchor-based approaches, which can be sensitive to anchor placement and scale selection. ViTs, on the other hand, can directly attend to relevant regions and learn to localize objects without the need for predefined anchors. This makes ViTs potentially more flexible and robust for object detection tasks.