1 Problem Statement

The problem being solved is generating captions for images. Using a Transformer model and the Flickr8k dataset, the goal is to caption an image. An image encoder and transformer decoder is used for this task. The Transformer is imeplemented, trained, tested and evaluated using the BLEU score.

2 Model Description

2.1 Model Architecture

The Transformer has:

Patch Size: **16**

Number of Encoder Layers: **4**

Number of Decoder Layers: **4**

Embedding Dimension: **512**

Feedforward Dimension: **512**

Number of Attention Heads: **8**

Patch Embedding:

In the Vision Transformer (ViT), the first step involves converting input images into a sequence of 2D patches. The patches are then flattened and linearly projected into embeddings. The patch size determines how the image is divided.

Positional Encoding:

Positional encodings provide the model with information about the relative or absolute positioning of patches in the sequence. Since the transformer does not inherently track the sequence order, this is important for maintaining sequence awareness.

Vision Transformer as Encoder:

The encoder uses a Vision Transformer (ViT) which is for processing images. After the patch embedding step, they are processed through multiple transformer encoder layers (4 in this case). Each encoder layer consists of multi-headed self-attention mechanisms and feedforward neural networks, which allows the model to focus on various parts of the image differently.

Decoder:

The decoder also has 4 layers, and includes self-attention and cross-attention that attends over the encoder's output. This structure helps the decoder focus on relevant parts of the input sequence.

Feedforward Networks:

Each layer in both the encoder and decoder contains a feedforward neural network with a dimension of 512. This network processes the data sequentially at each position across the sequence.

Attention Heads:

The model uses multi-head attention mechanisms with attention heads in both the encoder and decoder. Multiple heads allow the model to simultaneously attend to information from different subspaces at different positions.

Output Layer:

The final layer is a linear layer that projects the decoder output to the target vocabulary size, which is necessary for generating predictions.

2.2 Architecture Diagram

Used the diagram from this paper and annoted it to reflect the dimensions of transformer model used in this PA.

https://arxiv.org/pdf/1706.03762.pdf

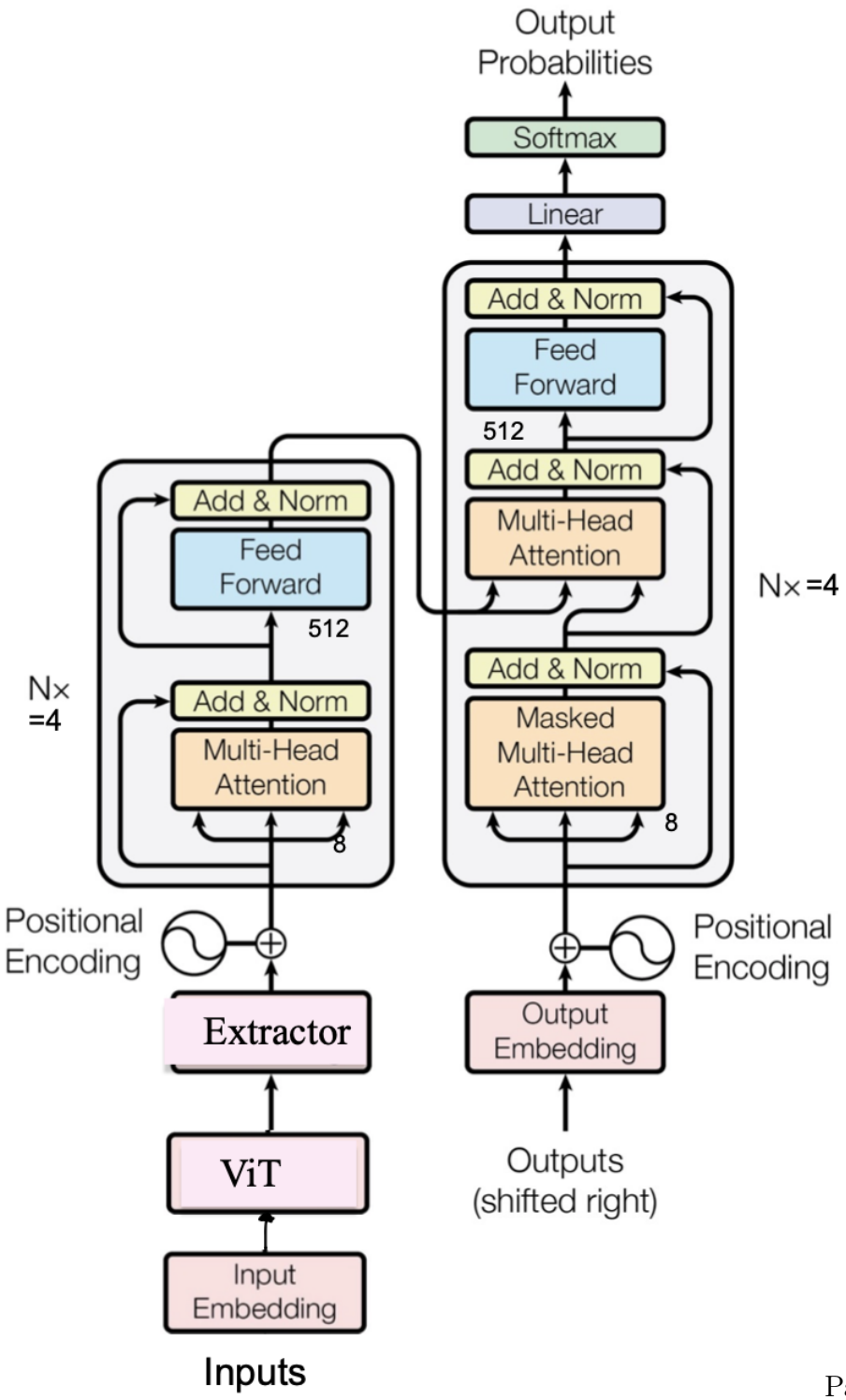

Figure 1: Transformer

3 Experimental Setup

3.1 Training/Testing

For the training, validation, and testing data, a batch size of 50 was used, which was determined after experimenting with different batch size values. 50 was a reasonable number as anything higher impacted the RAM while running.

3.2 Hyperparameters

The hyperparameters were determined by experimenting with different values.

- **Number of Epochs** = 15

- **Learning Rate** = 0.0001

- **Betas** = (0.9, 0.98)

- **Epsilon** = 1e-9

4 Experimental Results

4.1 Training and Validation Loss vs. Epochs

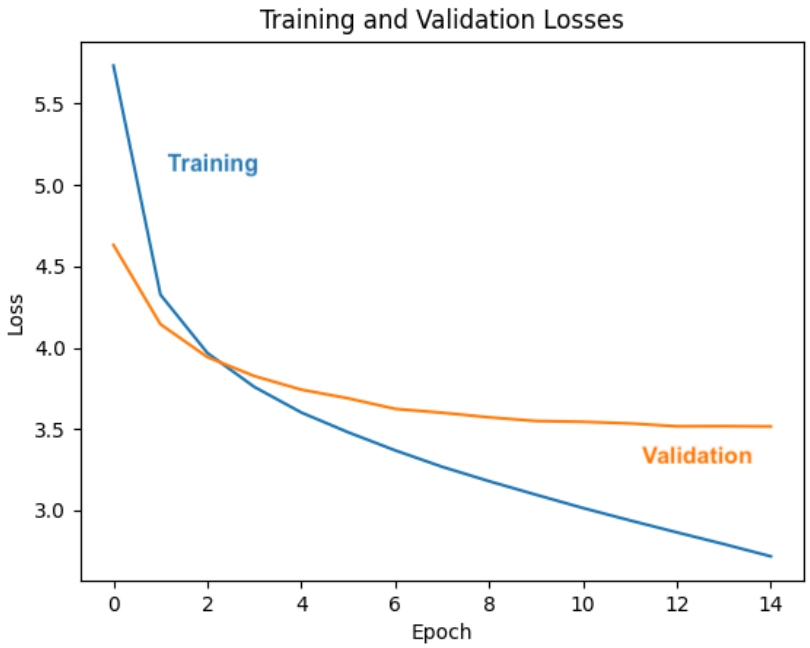

Figure 2: Training and Validation Losses

4.2 BLEU Scores vs. Epochs for Testing Data

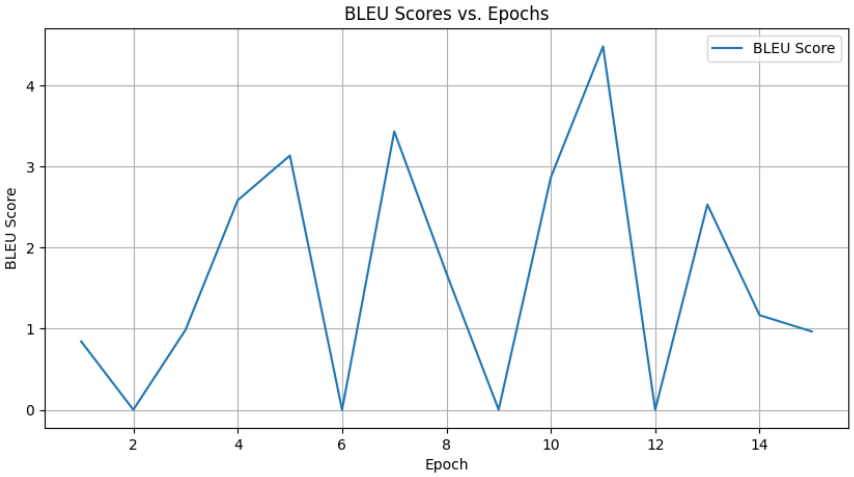

Figure 3: BLEU Scores

4.3 Final Results

The BLEU Score for the testing data is **5.94%**.

There are the losses and BLEU scores per epoch.

Epoch 1/15, Train Loss: 5.7331, Val Loss: 4.6313, Time: 18.37 sec

Average BLEU score: 0.8436900449896761

Epoch 2/15, Train Loss: 4.3260, Val Loss: 4.1443, Time: 18.35 sec

Average BLEU score: 0.0

Epoch 3/15, Train Loss: 3.9676, Val Loss: 3.9422, Time: 18.57 sec

Average BLEU score: 0.9857897733212235

Epoch 4/15, Train Loss: 3.7590, Val Loss: 3.8258, Time: 18.30 sec

Average BLEU score: 2.5857644746106727

Epoch 5/15, Train Loss: 3.6018, Val Loss: 3.7422, Time: 18.20 sec

Average BLEU score: 3.1377471854704155

Epoch 6/15, Train Loss: 3.4800, Val Loss: 3.6887, Time: 18.41 sec

Average BLEU score: 0.0

Epoch 7/15, Train Loss: 3.3691, Val Loss: 3.6240, Time: 18.42 sec

Average BLEU score: 3.435604720385129

Epoch 8/15, Train Loss: 3.2682, Val Loss: 3.6004, Time: 18.43 sec

Average BLEU score: 1.6843921482066324

Epoch 9/15, Train Loss: 3.1797, Val Loss: 3.5722, Time: 18.31 sec

Average BLEU score: 0.0

Epoch 10/15, Train Loss: 3.0964, Val Loss: 3.5496, Time: 18.48 sec

Average BLEU score: 2.8689770148576033

Epoch 11/15, Train Loss: 3.0148, Val Loss: 3.5448, Time: 18.35 sec

Average BLEU score: 4.484989424153506

Epoch 12/15, Train Loss: 2.9387, Val Loss: 3.5345, Time: 18.26 sec

Average BLEU score: 0.0

Epoch 13/15, Train Loss: 2.8652, Val Loss: 3.5171, Time: 18.36 sec

Average BLEU score: 2.533680173426038

Epoch 14/15, Train Loss: 2.7931, Val Loss: 3.5177, Time: 18.11 sec

Average BLEU score: 1.1675460082753841

Epoch 15/15, Train Loss: 2.7179, Val Loss: 3.5160, Time: 18.48 sec

Average BLEU score: 0.9662344351986855

5 Analysis

5.1 Comparing Different Architectures

All of the models below were run with the same hyperparameters mentioned in section 3.2. Unless it is the parameter being experimented with, the model hyperparameters are the same as section 2.1.

Table 1: Changing Encoder and Decoder Layers

Encoder Layers	**Decoder Layers**	**Training Loss**	**Validation Loss**	**BLEU**
4	4	2.7179	3.5160	5.944
3	3	2.8103	3.5245	5.252
3	4	2.7207	3.5051	3.505
4	3	2.8088	3.5234	4.2533

Among the different number of encoder and decoder layers, the training losses and validations losses have a small margin of difference. The chosen model, with 4 layers each, has the lowest for both. However, the BLEU score is the highest for the chosen model with a larger margin of difference. This shows that 4 layers is a good balance for this model's complexity. More than 4 layers was difficult to run.

Due to lack of GPU resources, I was unable to perform experiments with other variations of parameters. However, I did try to run the model with a patch_size of 8 and 32, both of which took too long to train, making 16 the most suitable number.

For some epochs, the BLEU score was 0 or quite low. This could be due to the model learning better for certain images and not others. Some common patterns I noticed were the words "man", "dog", and "shirt" being over used in the predictions. Most predicted captions had these words, which could be due to them being the most common words.

6 Conclusion

After successfully preprocessing the images and captions from the Flickr8k dataset, preparing them for training and testing with a Transformer model with a Vision Transformer (ViT), and evaluating its performance, a BLEU score of over 5% was reached.