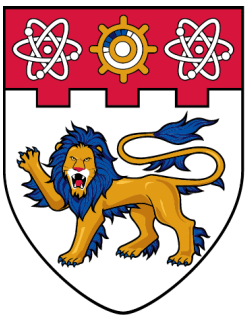


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

CE/CZ4042/SC4001 NEURAL NETWORK & DEEP LEARNING

Image Captioning with Neural Networks

Name	Matriculation Number
HENDY	U2122559J
Ng I-Shen Samuel	U2121307G

Table Of Contents

Problem Description 3

Background and Literature Review 4

Methodology 5

1. **Using Flickr 8k Dataset** 5
2. **Image Captioning with LSTM** 5
3. **Image Captioning with Transformer** 6
4. **Using BLEU score** 8

Experiments and Results 9

1. **Image Captioning with LSTM** 9
2. **Image Captioning with Transformer** 10

Conclusion 12

References 13

Problem Description

This project aims to create a system that predicts and generates descriptive text captions based on input images. The system implemented some of the concepts and models learned from the SC4001 Neural Network and Deep Learning module accompanied by research on how image captioning is being done. This project focuses on using and comparing LSTM and Transformer models to perform the task of robust image captioning and measure the performance of the models.

This project also describes the practical aspects of implementing LSTM and Transformer models and considers factors such as dataset selection, data processing, model architecture designs, hyperparameters tuning, and evaluation metrics selection. The project also highlights the challenges and benefits of using each model.

The solution of this image captioning system was done using Python Programming language to leverage powerful libraries and frameworks such as TensorFlow and Keras for deep learning tasks

Background and Literature Review

Image captioning through neural networks has witnessed significant advancements in recent years, and most of it is driven by the intersection of computer vision and natural language processing (NLP). Traditional methods, such as n-gram language models, were limited in capturing semantic richness. Deep learning, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs) like long short-term memory (LSTM), revolutionized image captioning.

The Transformer model that used the attention mechanism introduced in the paper "Attention is All You Need" by Vaswani et al. (2017), further revolutionized the field of natural language processing (NLP) by eliminating the need for recurrent neural networks (RNNs) and long short-term memory (LSTM) networks. It relies solely on self-attention mechanisms, making it highly parallelizable and efficient for processing sequences.

Transformer models have outperformed traditional CNN-LSTM architectures. Evaluating caption quality relies on metrics like BLEU (Papineni et al., 2002), which measures caption-reference overlap based on n-gram precision and brevity penalty.

Overall, CNN-LSTM and transformer models have elevated image captioning, enabling more accurate and contextually rich descriptions of visual content.

Methodology

1. Using Flickr 8k Dataset

Flickr 8k is a set of datasets that is publicly available and widely used in research and projects related to image captioning, it consists of a total of 8091 images in JPEG format and each image consists of 5 captions. Flickr also contains pre-split 6000 training, 1000 testing, and 1000 validation sets.

2. Image Captioning with LSTM

First, ResNet50 (a CNN pre-trained ImageNet dataset) was used to encode the input images. Then the image encodings/features were extracted from resnet50 and used as input to the Keras embedding layer.

The Keras embedding layer would then generate word embeddings on the captions associated with the images. Word embeddings represent words in a continuous vector space, allowing the model to learn meaningful representations of the words in the captions.

The embeddings were then passed into the LSTM model after which the image and text features were combined and sent to a decoder network to generate the next word.

Initially, the model was trained until 50 epochs, but there was overfitting. Therefore, another model was trained for 12 epochs, saving at each epoch, and the checkpoint where it had the best accuracy was selected.

The hyperparameters used were:
Epochs = 12
Batch size = 10
Text length = 15
The picture below shows the model layers:

Layer (type)	Output Shape	Param #	Connected to
input_layer_2 (InputLayer)	(None, 40)	0	-
input_layer_1 (InputLayer)	(None, 2048)	0	-
embedding (Embedding)	(None, 40, 256)	1,620,224	input_layer_2[0][0]
dropout (Dropout)	(None, 2048)	0	input_layer_1[0][0]
dropout_1 (Dropout)	(None, 40, 256)	0	embedding[0][0]
dense (Dense)	(None, 256)	524,544	dropout[0][0]
lstm (LSTM)	(None, 256)	525,312	dropout_1[0][0]
add (Add)	(None, 256)	0	dense[0][0], lstm[0][0]
dense_1 (Dense)	(None, 256)	65,792	add[0][0]
dense_2 (Dense)	(None, 6329)	1,626,553	dense_1[0][0]

Total params: 4,362,425 (16.64 MB)
Trainable params: 4,362,425 (16.64 MB)
Non-trainable params: 0 (0.00 B)

Figure 1. LSTM model

The model was then tested during the caption generation by using greedy search and beam search. Greedy search is a decoder used to select the highest probability at each step and append it to the caption being generated, while beam search is a decoder that considers multiple candidate captions simultaneously. In conclusion, greedy search provides a quick and straightforward approach to caption generation, while beam search offers improved caption quality by exploring a wider range of possibilities.

3. Image Captioning with Transformer

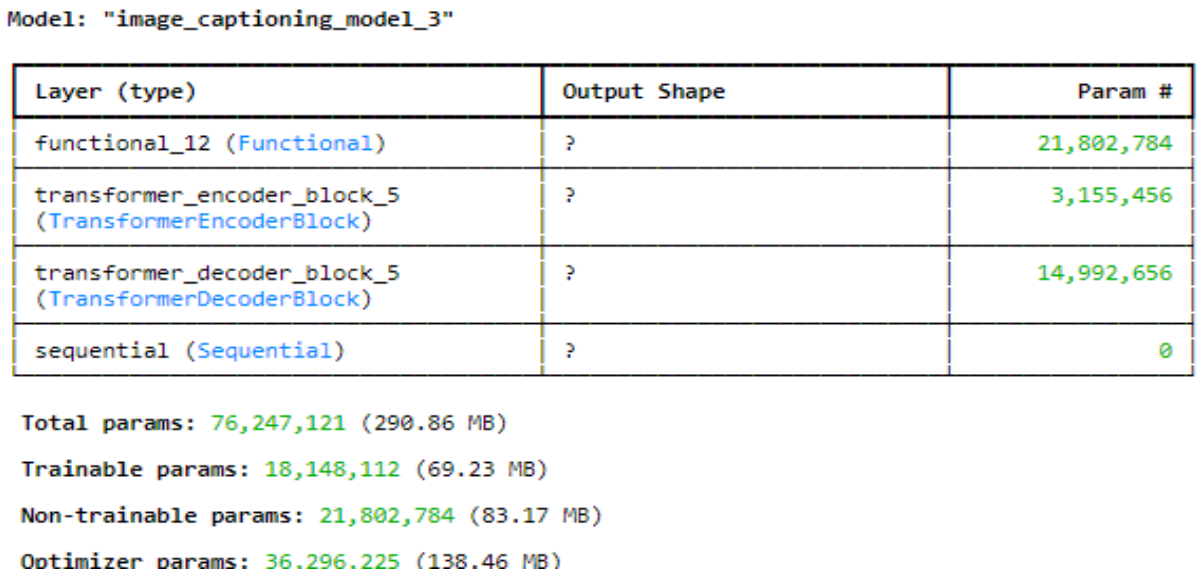
Image captioning using a transformer model relies on self-attention mechanisms to capture global dependencies within input sequences. The advantage of using the transformer model is that it can handle long-range dependencies and capture global context without recurrence.

For this project, Flickr 8k images and texts were loaded and split into 80% for training (6114 images), 10% for testing (764 images), and 10% for validation (765 images) instead of using the pre-split dataset to allocate more training samples to yield a better result. Important parameters were also set to epochs = 10, batch size = 64, vocabulary size = 10000, and output text length = 25.

Based on Vaswani et al. (2017), transformer models typically have a larger number of parameters compared to traditional models like CNN-LSTM. With more parameters, transformer models have a higher capacity to learn complex patterns and relationships in the data. Therefore, providing more training data allows the model to effectively utilize its capacity and learn more robust representations.

After the splitting, the texts would be preprocessed to remove unwanted characters and converted the text data into numeric vectors using TensorFlow's TextVectorization for the transformer model to process the numeric vector efficiently. Transformers process all tokens in the input sequence simultaneously using self-attention mechanisms. This allowed for the parallel processing of input tokens and enabled the model to capture long-range dependencies more effectively which is different from LSTM which generates the token based on the previous token and visual features

The model and layers were then defined as shown in the picture below.



Model: "image_captioning_model_3"

Layer (type)	Output Shape	Param #
functional_12 (Functional)	?	21,802,784
transformer_encoder_block_5 (TransformerEncoderBlock)	?	3,155,456
transformer_decoder_block_5 (TransformerDecoderBlock)	?	14,992,656
sequential (Sequential)	?	0

Total params: 76,247,121 (290.86 MB)

Trainable params: 18,148,112 (69.23 MB)

Non-trainable params: 21,802,784 (83.17 MB)

Optimizer params: 36,296,225 (138.46 MB)

Figure 2. Transformer model

The model consists of :

1. **CNN layer** was used for extracting visual features from the input images
2. **Transformer encoder** applied self-attention that converts different parts of visual input and encodes it into a meaningful representation
3. **Transformer decoder** generated captions based on the encoder output
4. **The sequential layer** was used for assembling the components of the model

The base model utilized pre-trained InceptionV3 models for extracting image features and the Input images passed through a CNN layer to extract its visual features that capture the essential visual information of the image after the transformer encoder layer applies self-attention to effectively capture dependencies within the visual input, self-attention enables the model to focus on different parts of the image, allowing for more comprehensive understanding and representation of visual content which will result into meaningful representation for decoder to generates captions based on the encoder output. The decoder utilizes the attention mechanism to attend to relevant parts of the input image features and generate descriptive captions accordingly.

In the decoder, two components were introduced, mainly positional embedding which adds positional information to the input embeddings, combining token embeddings with positional embeddings to incorporate sequence order into the input representations. The other component is the causal attention mask that prevents the decoder from attending future tokens during the training process, ensuring autoregressive caption generation. These two attributes of transformer models have a large number of parameters compared to the LSTM model.

The model architecture was assembled using a Sequential layer, which combines the components of the CNN layer, transformer encoder, and transformer decoder sequentially. This structured approach ensures that the visual features are appropriately processed and utilized by the transformer architecture for accurate caption generation.

The model was then tested and the result was compared with its true caption provided by the dataset based on the BLEU (Bilingual Evaluation Understudy) standard.

4. Using BLEU Score

BLEU (Bilingual Evaluation Understudy) is used in this project to evaluate the performance of the LSTM and transformer model in performing image captioning. BLEU is widely used and has been adapted to evaluate image captioning systems based on its ability to assess the similarity between the generated text and the reference text. BLEU measures the similarity between two sentences and is expressed on a scale from 0 to 1, where scores typically peak realistically between 0.6 and 0.7.

In several studies, BLEU has been used as a general criterion for assessing the task of image captioning. For instance, BLEU was used to evaluate the performance of their model for image captioning in a paper called "Show and Tell: A Neural Image Caption Generator" by Vinyals et al. 2015. Similarly, the paper "Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering" by Anderson et al. (2018) employed BLEU to assess the quality of generated captions.

Calculation of the Bleu score involves two main components: N-gram and Brevity Penalty. N-gram evaluates the number of matching consecutive sequences of N words between the generated and reference sentences. Commonly, N values of 1, 2, 3, and 4 are employed, and their respective scores are averaged. To prevent inflated scores caused by repeated sequences in the prediction (such as 'a a a a a a' or 'a dog a dog a dog' matching 'a dog is in a field'), the count for each sequence is capped at the maximum occurrence in the reference sentence.

The Brevity Penalty addresses the issue of overly short predictions, such as single-word responses like 'the,' by penalizing scores for predictions shorter than the reference sentence. The final Bleu score is derived by multiplying these two scores together.

To compare our models, we select 10 images, using both models to predict captions for them, calculating bleu scores, and comparing the averages.

Experiments and Results

This section discussed the performance of LSTM and Transformer models in the task of image captioning. The primary objective was to evaluate the effectiveness of each model architecture in generating descriptive captions for input images. The experiments were designed to assess various aspects of model performance, including caption quality, computational efficiency, and scalability.

1. Image Captioning with LSTM

The model using LSTM obtained an average bleu score of 0.1088 when using beam search and the model trained on 9th epochs.

Below are the Training and Validation accuracies plotted against epochs.

It seems that overfitting has occurred, with the loss of the validation set increasing while the loss of the training set decreases. Epoch 9 has the highest accuracy on the validation set, so it will be used for further evaluation.

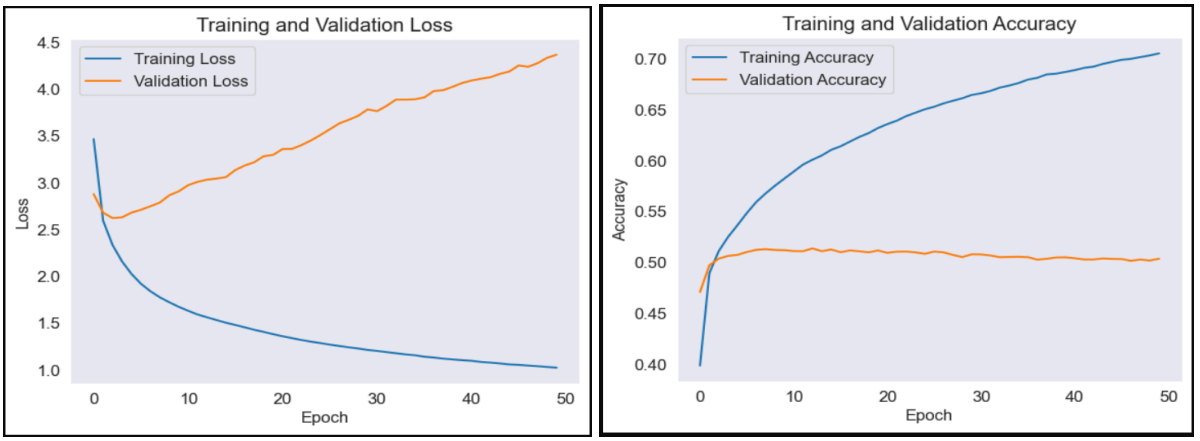


Figure 3a. Loss and Accuracy for 50 epochs

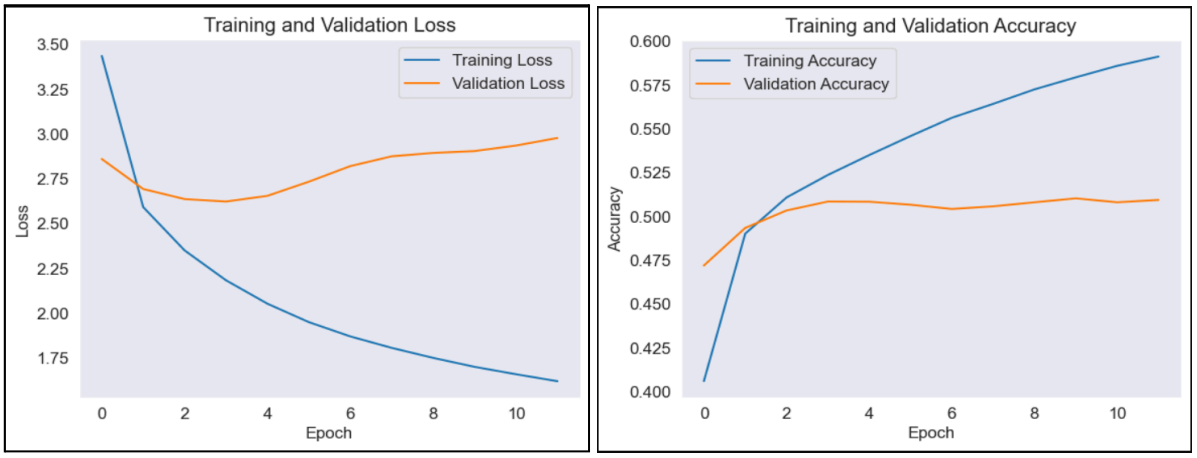


Figure 3b. Loss and Accuracy for 12 epochs

The poor loss and accuracy chart could be caused by the possibility that the model may not have been able to effectively capture the complex relationships between the input images and their corresponding captions. This could be due to the limitations of the LSTM architecture in handling long-range dependencies or the lack of sufficient training data to adequately generalize to unseen examples.

Below are examples of captions generated by the trained model.

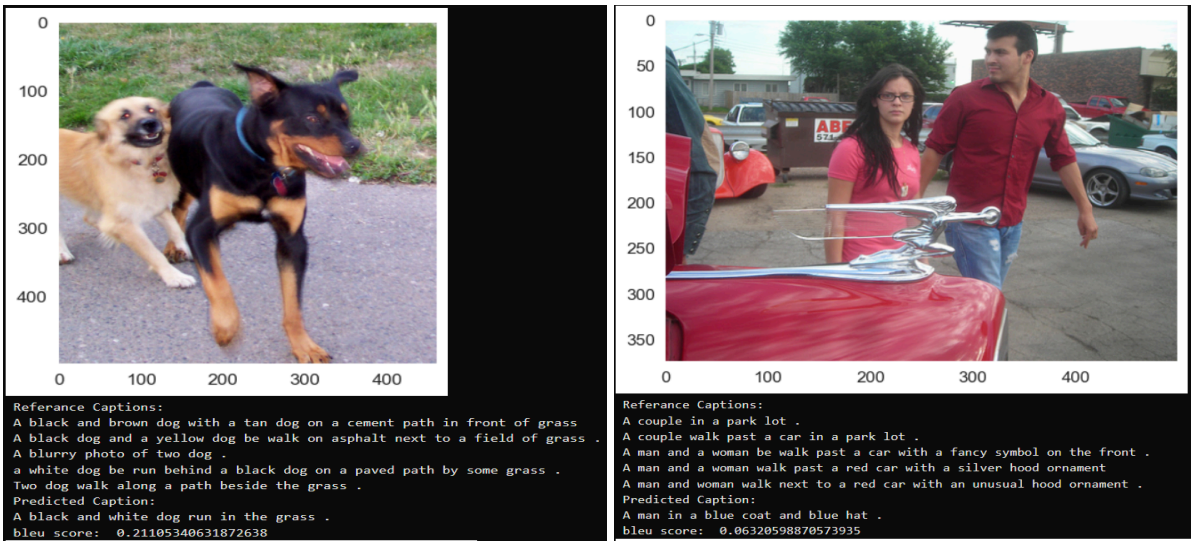


Figure 4. Generated Captions

For 10 selected images, an average bleu score of 0.172 was achieved.

2. Image Captioning with Transformer

The results obtained with the Transformer model after running 10 epochs showed an increase in accuracy and a decrease in loss for both training and validation (shown in Figure 5). The trained model exhibited improved performance indicating its robustness in learning visual features and generating coherent captions.

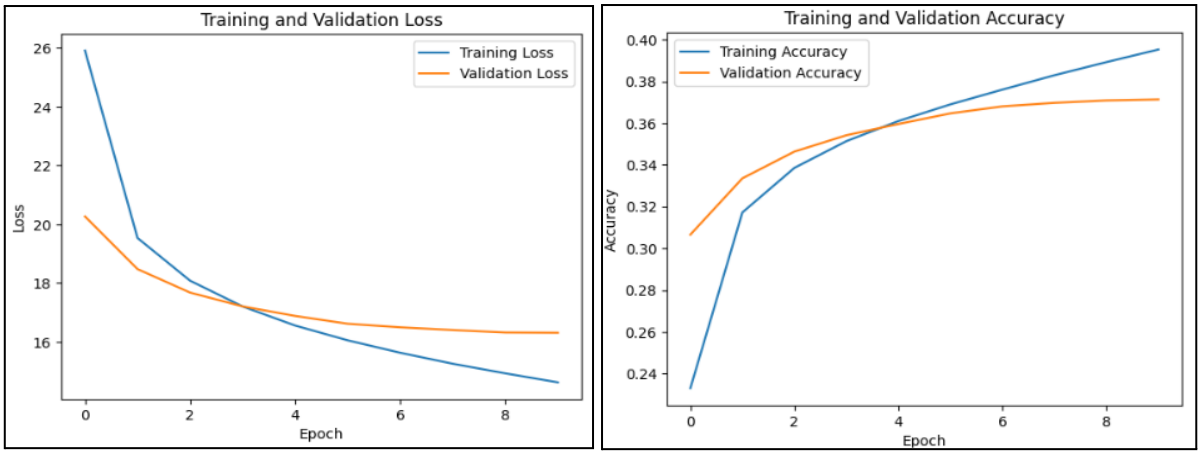


Figure 5. Loss and Accuracy for 10 epochs

Here are some of the predicted captions generated using the trained Transformer model on images from the test dataset. Good results, shown in Figure 6, achieved BLEU scores of 0.989 and 0.844, while poor results, shown in Figure 7, obtained BLEU scores of 0.237 and 0.177, and the predicted captions generated were significantly different from the original captions of the images.

3625957413_e475943aa3.jpg
True Caption: ['A black and white dog is running along side a small brown one .', 'The dogs run in the green field .', 'Two dogs runnin

Predicted Caption: a black and white dog is running through a field

BLEU Score: 0.9890654562541585

Images ID = 3651971126_309e6a5e22.jpg
True Caption: ['A black and brown dog with a tan dog on a cement path king on asphalt next to a field of grass .', 'A blurry photo of two dc ved path by some grass .', 'Two dogs walk along a path beside the gras

Predicted Caption: a black and white dog is running through the grass

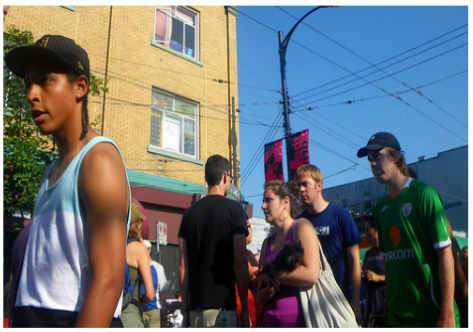
BLEU Score: 0.8443258653392445

Figure 6. Good Result of the Prediction

3639105305_bd9cb2d1db.jpg
True Caption: ['A mother and daughter pose in front of a waterfall e with a waterfall in the background .', 'A woman and a girl sitti hter in front of a waterfall', 'A woman is sitting with a girl on

Predicted Caption: a dog is running through the grass

BLEU Score: 0.176992164297146



True Caption: ['A crowd of people are walking down a city street .', 'A group a peo ding .', 'A group of people stands in front of a yellow building .', 'People gather a brick building .']

Predicted Caption: a man in a red shirt and white shirt is standing on a skateboard

BLEU Score: 0.3179400864727676

Figure 7. Poor Result of the Prediction

The poorer results could be attributed to the smaller training dataset size, which may have made some features unrecognizable. However, it is worth noting that setting a higher training size might lead to overfitting and require additional computational resources during both training and inference, particularly due to the transformer model's parallel processing nature.

For 10 selected images, an average bleu score of 0.611 was achieved.

Conclusion

This project explored the efficiency of LSTM (Long Short-Term Memory) and transformer-based models for image captioning.

The LSTM model offers a simpler and more interpretable alternative. it demonstrated strong performance in generating captions for images. Its sequential processing nature allows it to effectively capture temporal dependencies within input sequences, making it well-suited for tasks involving sequential data like natural language processing. However, the LSTM model's ability to capture long-range dependencies may be limited by the vanishing gradient problem, which could affect its performance on longer sequences.

The transformer-based model offers the ability to handle long-range dependencies and capture global context without recurrence, which is crucial for understanding the visual content of images and generating coherent captions. In Addition, the transformer model's parallel processing architecture makes it highly scalable and efficient, particularly for tasks involving large datasets. Transformer also mitigates some challenges associated with vanishing gradients that occur in the LSTM model.

By comparing the results of the performance, we conclude that the transformer model performs better compared to LSTM. While LSTM models might excel in temporal dependencies and sequential information capture, Transformer Models leverage attention mechanisms to effectively process long ranges of dependency and global context. However, the Transformer's parallel processing architecture may contribute to its computational efficiency and scale.

In conclusion, The Transformer model showed a better result and was the preferred choice for image captioning tasks, offering valuable insights for future research in this domain.

References

Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on Association for computational linguistics (pp. 311-318).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).

AmritK10. (n.d.). Amritk10/image_captioning: Image captioning using LSTM and deep learning on flickr8k dataset [GitHub repository]. Retrieved from https://github.com/AmritK10/Image_Captioning/tree/master.

Doshi, Ketan. (2021, May 11). Foundations of NLP explained - bleu score and WER metrics. Medium. https://towardsdatascience.com/foundations-of-nlp-explained-bleu-score-and-wer-metrics-1a5ba06d812b.

Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and Tell: A Neural Image Caption Generator. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., & Zhang, L. (2018). Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).