CS 4782 Final Report

Sunny Sun, Michael Wei, Tony Chen, Linda Hu, Jiye Baek

May 5, 2025

1 Introduction

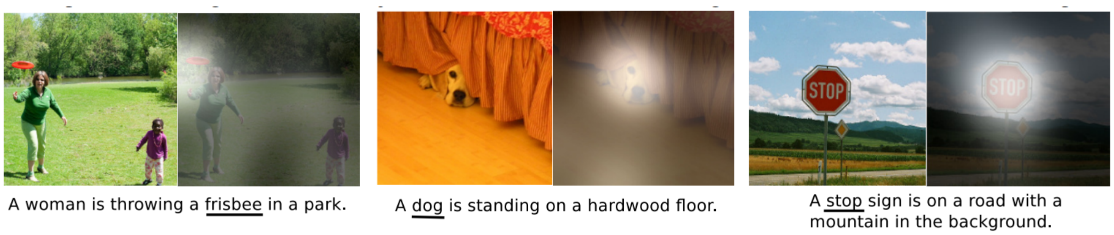
In this project, we aimed to reproduce the image captioning results of the paper *Show, Attend, and Tell* published in 2016 by Kelvin Xu *et al.* This paper utilizes the then novel technology of attention to create more accurate and efficient methods of automatic caption generation of a given input image. This paper was motivated by the computer vision problem of determining not only what objects exist within an image, but also accurately describing the relationships between them in a natural language.

Although prior image-captioning networks existed at the time this paper was written, most used static global image features which often miss fine-grained spatial details crucial for accurate captioning. This paper's main contributions include introducing two attention-based image caption generators under a common framework, visualizing the results of this framework by outputting alpha masks that depict the area of an image that the attention focused on, and quantitatively validating the usefulness of attention in caption generation using standard metrics of BLEU and METEOR scores.

2 Chosen Result

Besides implementing the underlying infrastructure of the model outlined in the paper, we aimed to reproduce two main results:

Visualization of soft attention at each timestep of a generated caption. The original paper has several examples of the change in attention over time to reflect the relevant parts of the image. We felt that this visualization was rather significant in understanding the attention mechanism used by the model and to gain intuition into what the model saw.



A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

A stop sign is on a road with a mountain in the background.

Figure 1: Visual examples of soft attention from the original paper

BLEU and METEOR scores reported for the soft attention model on the Flickr8k dataset. These are common metrics used in natural language processing that quantitatively evaluate the quality of generated captions against human understanding. We felt that these results from the paper were significant in providing a concrete standard from which we can measure the accuracy of our model.

3 Methodology

The model takes in a single raw image and passes it through a convolution neural network (CNN) encoder such as VGG, ResNet, or DenseNet to extract spatial image features in annotation vectors. Instead of training

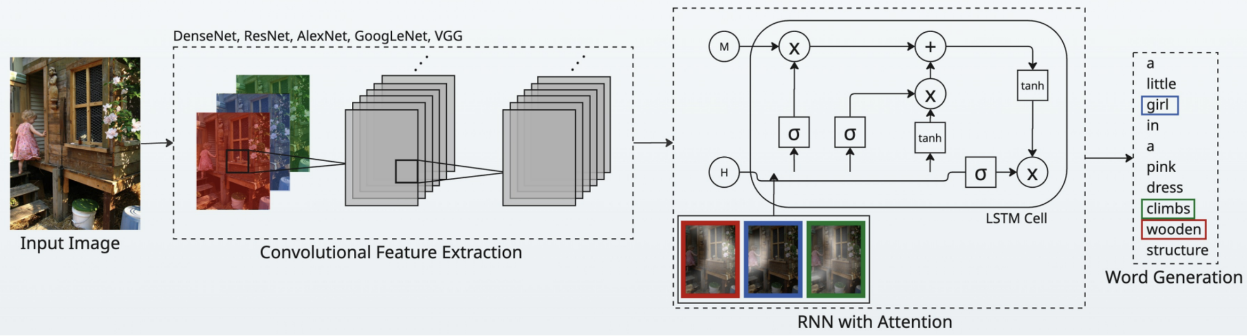


Figure 2: Overall Model Architecture

from scratch, we used frozen, pre-trained CNNs encoders, VGG16, AlexNet, DenseNet121, ResNet50, and GoogleNet, to experiment with newer architectures and save compute.

The attention model calculates the weights for each annotation vector. These are used in attention mechanism to generate a context vector, indicating where in the image the model attends to at that timestamp. The paper uses two types of attention: hard and soft attention. Soft attention calculates the expected value of all annotation vectors using the softmax of the weights, whereas hard attention uses the weights for each annotation vector to create a multinoulli distribution which is then sampled. Then, hard attention samples from each location of the image as one-hot variable–1 if the i-th location of the image is used to extract visual features, else 0. Thus, the context vector in the hard attention implementation is a multinoulli distribution. We ultimately chose to omit the implementation of hard attention due to its reliance on reinforcement learning, which introduces additional complexity and instability during training.

In the LSTM decoder, we initialized the memory state and hidden states by taking the average of the annotation vectors fed through 2 MLPs. The context vector, previous word, and previous hidden state are fed into LSTM decoder at every word predicted in the caption.

The original paper trained the soft attention model on MS COCO for 3 days on an NVIDIA Titan Black GPU. Due to time and memory constraints, we only trained our re-implemented model on the Flickr8k dataset alone, avoiding Flickr30k and MS COCO in the paper. The smaller dataset also allowed us to maintain fast experimentation cycles and train our model in around 2 hours. In the dataset, there are 5 human annotated captions per image. To evaluate our caption quality, we used BLEU and METEOR scores to calculate the training and validation accuracy. We calculated these scores for soft attention alone, but across all 5 pretrained CNN encoders.

4 Results and Analysis

In terms of BLEU and METEOR scores, our re-implementation achieved slightly lower compared to the original paper's soft attention scores for the Flickr8k dataset. We also trained the model with various frozen, pre-trained encoders and found that ResNet performed the best. However, in the original paper, the model fine-tuned its encoders during training. We believe that this may account for the discrepancy between our model's and the original paper's BLEU and METEOR scores. The scores of our models and the original paper are shown in the table below.

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR
Soft-Attention (Original Paper)	67.0	44.8	29.9	19.5	18.93
AlexNet	60.44	34.90	19.71	10.81	12.89
DenseNet	64.81	40.82	26.26	15.84	16.95
GoogLeNet	64.53	40.13	25.12	14.77	16.45
VGG	64.79	40.37	25.88	15.48	17.08
ResNet	**65.67**	**41.40**	**27.13**	**16.48**	**17.94**

In terms of visualization and qualitative performance, we fed several random images to our model and obtained captions and attention maps after training. We can see in words such as "drive" and "is" that the model is able to attend to salient regions for even words that are non-object. The model learns alignments that correspond pretty strongly to human intuition, similar to the original paper's findings. Based on the generated attention masks, we would claim that our re-implemented model is capable of building a coherent visual-linguistic representation, grounding each word in relevant spatial context.

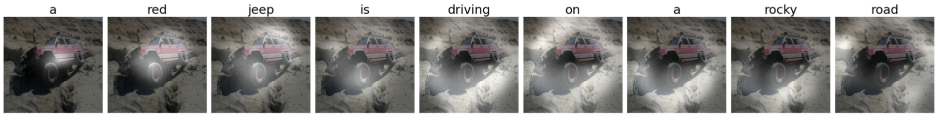


Figure 3: Example of our model's attention mechanism visualization per word in generated caption.

One challenge that we faced in the re-implementation process was long training times and timing out on compute while using a T4 GPU on Google Colab. On average, training 10 epochs would take almost two hours. We partially resolved these problems by optimizing the decoder of the model. We also implemented saving model checkpoints at every epoch so we could break up training time over multiple sessions.

5 Reflections

Our re-implementation demonstrated meaningful results given limited resources. Despite only training on the smaller Flickr8k dataset and using frozen pre-trained CNN encoders, our model was still able to generate reasonable captions and produce interpretable attention maps. This highlights the effectiveness of the soft attention mechanism, even under constraints.

One of the key lessons we learned was the importance of feasibility given time constraints and tight resources. Although the original paper explored both hard and soft attention mechanisms, we found that soft attention was significantly more practical to implement and train due to its differentiable and deterministic nature. Implementing the hard attention, especially the required reinforcement learning techniques like REINFORCE, from scratch was more time consuming than anticipated.

Overall, this project deepened our understanding of attention mechanisms in image captioning and provided practical experience in re-implementing a significant paper in computer vision and natural language processing. For potential future work and with more time and computational capacity, we would look into using larger datasets that the original paper used such as Flickr30k and MS COCO to improve generalization and caption quality. We can also try to implement hard attention mechanism using REINFORCE to explore sharper spatial focus. Finally, we can fine-tune CNN encoder during training to allow the model to learn image features more tailored to caption generation.

6 References

[TODO DO WE NEED TO CREDIT MORE TOOLS ??]

- Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Rich Zemel, and Yoshua Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. 2015. Paper Website

- Kelvin Xu. Arctic Captions: Theano Implementation of Show, Attend and Tell. GitHub repository. https://github.com/kelvinxu/arctic-captions

- Aaron C.C. Wong. Show, Attend and Tell: PyTorch Implementation. GitHub repository. https://github.com/AaronCCWong/Show-Attend-and-Tell

- Hodosh, M., Young, P., & Hockenmaier, J. (2013). Framing Image Description as a Ranking Task: Data, Models and Evaluation Metrics. Journal of Artificial Intelligence Research, 47, 853–899. https://doi.org/10.1613/jair.3994

- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... & Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Advances in Neural Information Processing Systems (NeurIPS 2019), 32.