Augmentation and Adjustment: Improving Neural Image Captioning Accuracy

Mary Dao, Kevin Russell, Tarandeep Singh

I. Introduction

The intersection of vision and natural language processing has been a widely active topic in the machine learning world in the past 7 years. With the Neural Image Caption (NIC) technique introduced in 2015, Vinyals and his team were able to achieve state-of-the-art performance in this image representation task [1]. Although the development of this neural network architecture was a successful and extraordinary feat, there were still many mislabeled images shown and discussed in the paper. For the visually impaired, who depend on this type of technology, incorrect image captions could mislead the actual situation. The question then becomes whether or not the performance for NIC can be improved.

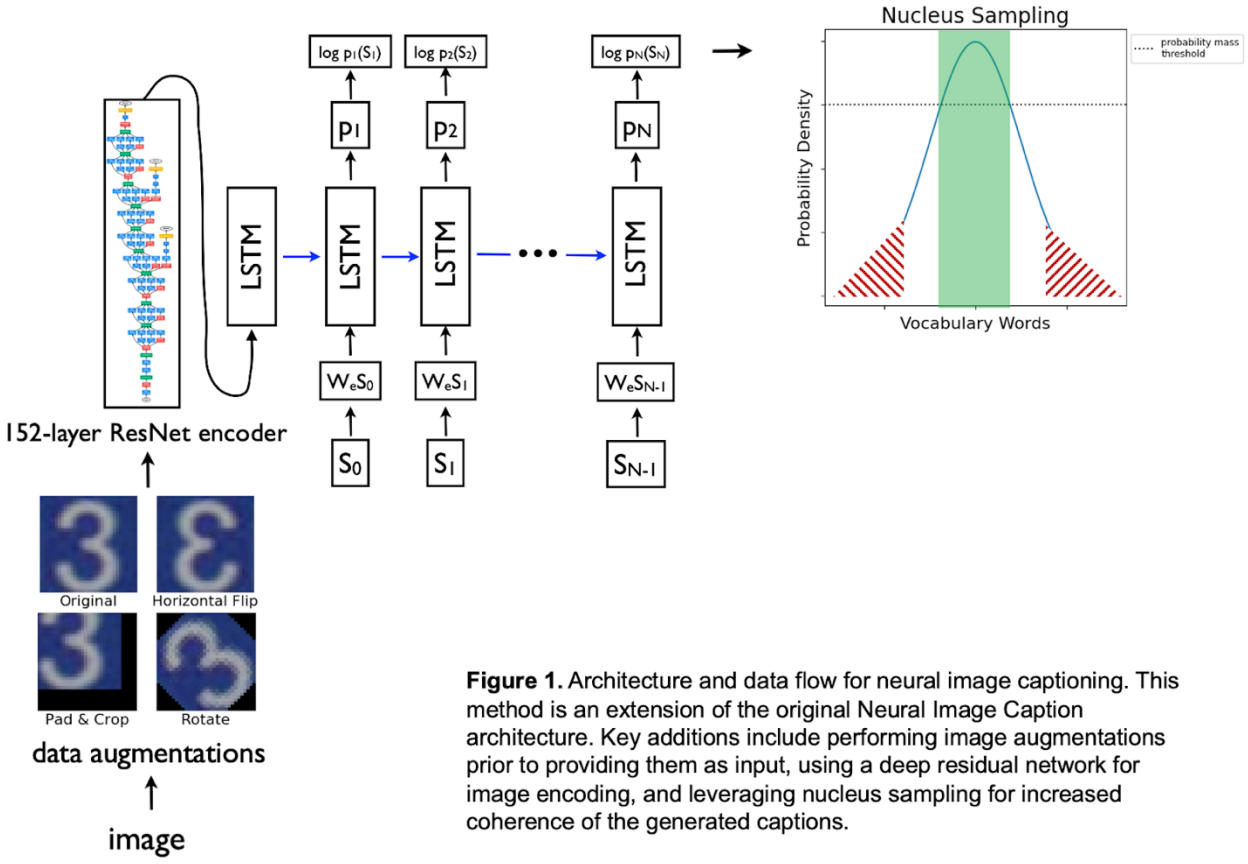

Figure 1. Architecture and data flow for neural image captioning. This method is an extension of the original Neural Image Caption architecture. Key additions include performing image augmentations prior to providing them as input, using a deep residual network for image encoding, and leveraging nucleus sampling for increased coherence of the generated captions.

Our goal is to improve its performance in order to better serve those who need it. We set out on this task by taking some learnings from the Deep Learning course by Professor David Bau, the author's own recommendations, and following our own intuition. Over the course of the project, we built our own NIC from scratch. In doing this, we made some modifications and upgrades to its existing architecture, the training data, and its evaluation criteria (Figure 1). Our results show that NIC can be improved significantly and additionally, we identified some methods that can be incorporated to improve NIC further.

II. Background

Since Yann Lecun's invention of convolutional neural networks (CNNs) in 1989 [2], many have adopted deep learning models for processing images. Previously proposed solutions for image captioning tended to stitch together multiple frameworks [3, 4]. The significance of the *Show and Tell* paper lies in its end-to-end system design, which leverages both deep CNNs for image processing and Recurrent Neural Networks (RNNs) for sequence modeling to create a single CNN-RNN network that generates descriptions of images. They were heavily inspired by Cho, et. al. (2014), who pioneered the dual RNN-RNN encoder-decoder structure that achieved state-of-the-art performance in machine translation [5].

This approach allows for image processing and text generation to occur within the same network, in contrast to methods such as the one developed by Li, et. al., which began with detections and pieced together a final description using phrases containing detected objects and relationships [3]. With all of these endeavors to solve a task that is simple for humans but extremely complex for machines to do, we were motivated to dissect the NIC model and discover how we could make additional improvements without completely changing the overall structure.

III. Methods

For our project we chose to build the method for NIC from scratch, and to add a deeper 152-layer Residual Network (ResNet152), a Layer Norm between the Long-Short Term Memory (LSTM) cells, several different image augmentations, and nucleus sampling. We wanted to understand the architect so that we can understand the best way to improve its performance. In total, we compared results for the original NIC paper architect, Shwetank Panwar's PyTorch-based NIC architect [6], and the re-implementation of our own version of NIC (Table 1).

Table 1. Architecture comparison of the original NIC model, Shwetank Panwar's PyTorch implementation of NIC, and our own PyTorch implementation of NIC.

NIC (2015)	Shwetank (2020)	NIC (2022 Re-implementation)
• CNN encoder	• ResNet50 encoder	• Deeper ResNet152 encoder
• Beam Search	• Top sampling	• Nucleus sampling
	• Simple image augmentations	• Image augmentations (simple and complex)
• LSTM per word	• LSTM called once	• LSTM per word
• Dropout	• Dropout between LSTM and Linear	• LayerNorm between LSTM and Linear

For augmentations, we chose three approaches that used simple and complex transformations, which are summarized below in Figure 2. We used the Albumentations Python package [7], which boasts fast and flexible image augmentations, to perform these transformations. We selected these functions because we wanted to create diversity within the dataset so that the model could better learn image captioning.

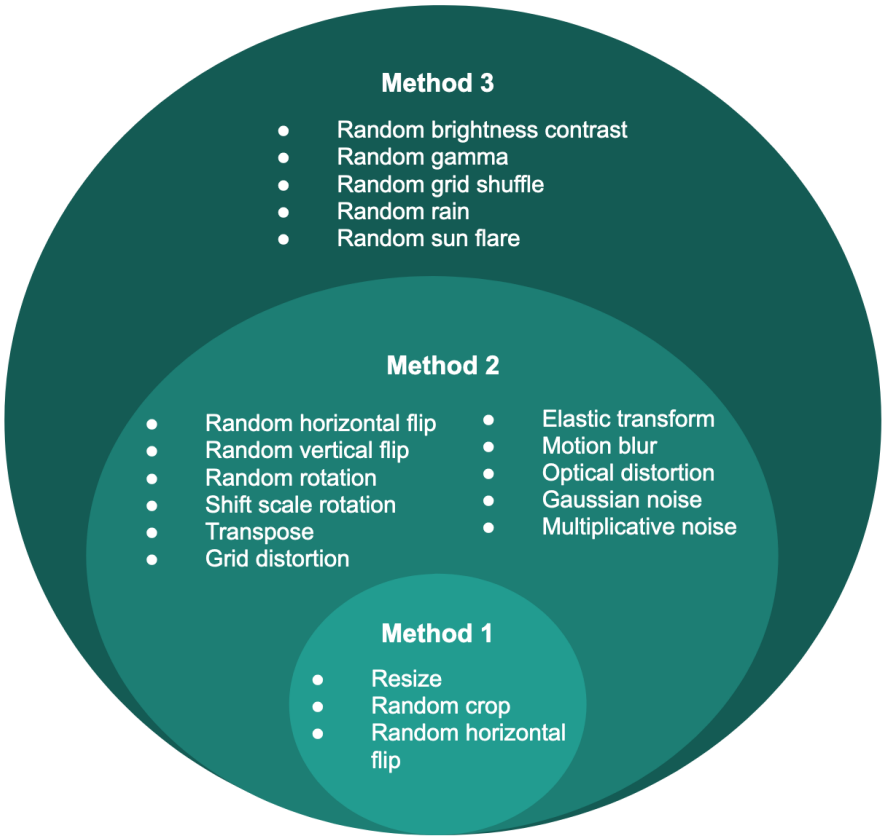

Figure 2. Summary of different image augmentations applied on input images before feeding to the NIC model. Each method employs listed augmentations in addition to those from all previous methods.

We also discovered a newer sampling method called nucleus sampling that leverages probability mass to filter the cumulative distribution function (CDF) of the word probabilities to sample among words that may be more surprising than samples made through beam search or top sampling [8]. To implement nucleus sampling, we chose to utilize Temperature Sampling to choose among the word probabilities. Temperature Sampling is inspired by statistical thermodynamics where high temperature means low energy states are more likely to occur [9].

With this new sampling technique, we are able to generate more coherent captions that are more human-like. The main problem we faced with this method was how to tune the probability mass to get the most accurate captions. We decided to select a batch of 100 samples and test both the BLEU-1 and BLEU-4 scores against different probability mass numbers. We observed that a probability mass of 13 is the optimal value according to our tests (Figure 3) and continued with that value in our experiments thereafter.

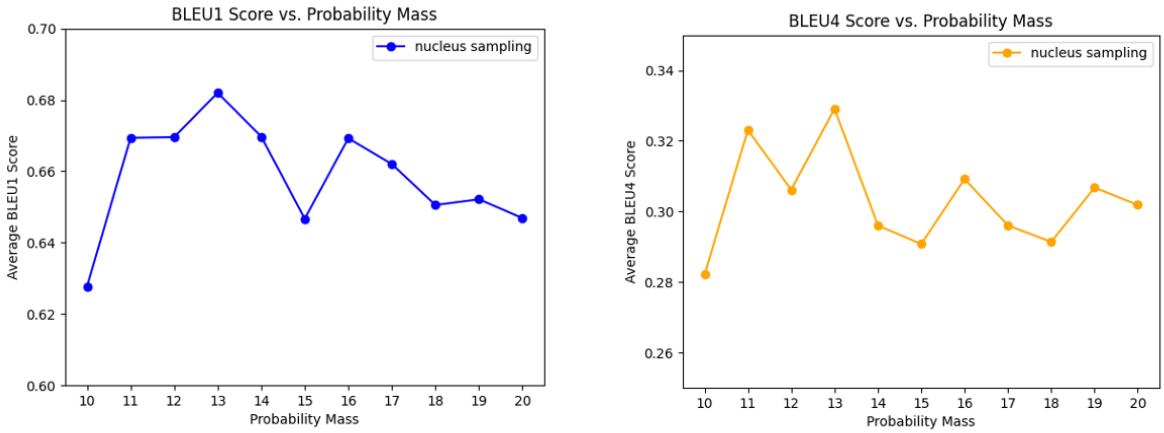

Figure 3. BLEU-1 and BLEU-4 scores as a function of probability mass threshold value used for nucleus sampling during caption generation.

IV. Results

Unfortunately, our Methods 2 and 3 for image augmentations resulted in suboptimal image captions, but we were able to generate text descriptive of the input images using Method 1 (Figure 4). As shown in Figure 4 (A), the captions on both images do not describe the scene at all. We suspected that this is due to the more complex image augmentations performed on the images for the second and third methods.

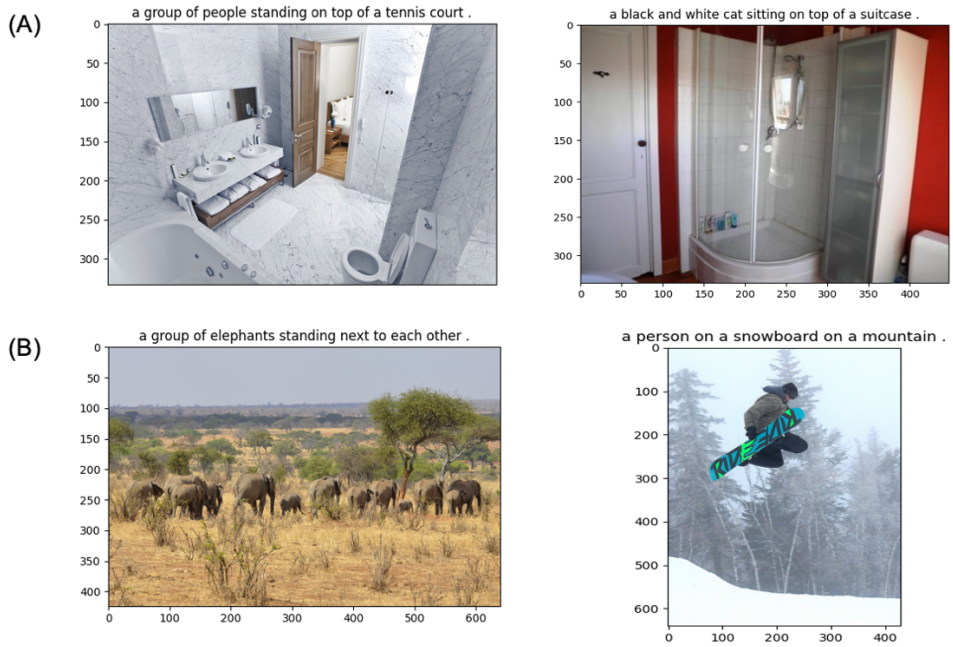

Figure 4. Predicted image captions generated using trained NIC-based models. (A) Text description of images from models that performed complex image augmentations on the input images prior to encoding them. (B) Captions outputted from models that only performed simple augmentations

By probing into the model, we discovered the underlying reason these methods were underperforming. The sample augmentation image displayed in Figure 5 confirmed our belief that image augmentation can be detrimental to the model when applied in excess. In identifying this issue with Methods 2 and 3, we decided to move forward using Method 1 only to train the model and then test the effect of nucleus sampling.

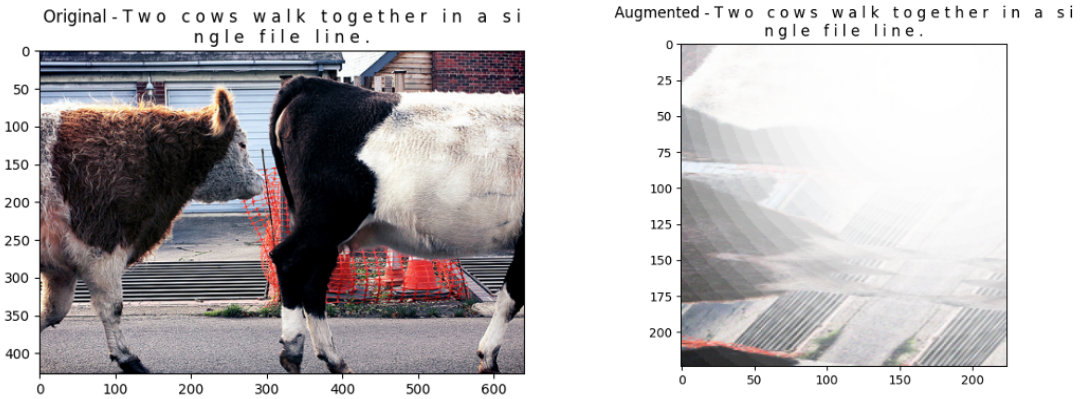

Figure 5. Comparison of a sample input image and the resultant image under complex augmentation.

We evaluated the model by generating captions for every image in the validation set of 5,000 images, computing the BLEU-4 scores against the five human-written annotations as the reference, and taking the mean of all scores. The results show that our model outperforms both the original NIC architecture and the aforementioned Shwetank model (Figure 6).

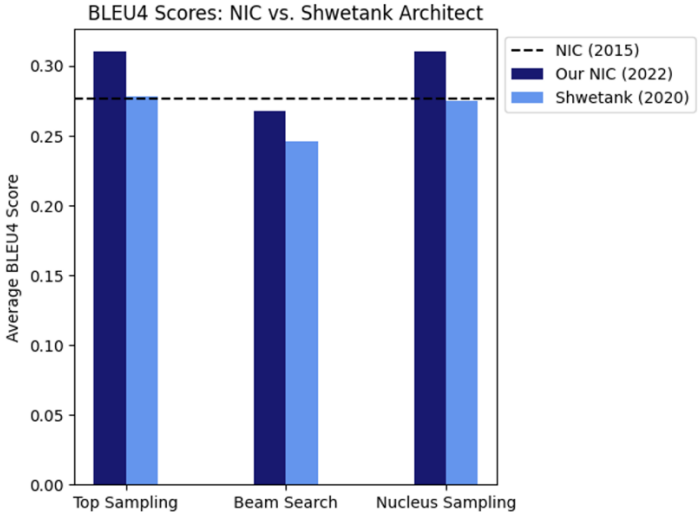

Figure 6. Average BLEU-4 scores on validation dataset (n = 5,000) for each sampling method and each implemented architect. Dashed horizontal line indicated BLEU-4 score reported in the original NIC paper.

Notably, the use of nucleus sampling significantly increased performance in both of the tested models, yielding BLEU-4 scores of 31.1 and 27.5 for our model and Shwetank's model, respectively. When generating captions with beam search, however, this degraded the scores by approximately 3 points for both methods. Nucleus sampling is a relatively new concept that was published 5 years after the original NIC *Show and Tell* paper, and our results demonstrate that this technique is more effective than beam search.

Regardless of the sampling technique selected for the caption generation, our architecture performs better than the Shwetank model in terms of BLEU-4 scores. When employing nucleus sampling, for example, our model evaluation metric exceeds Shwetank's by about 4 points. This could be due to our choices for data augmentation, the increase in ResNet hidden size, the use of LayerNorm instead of dropout, or the number of calls to the LSTM decoder. Due to a lack of computing power, we did not have the chance to train separate models and identify the most effective architecture adjustment. However, this would be an interesting aspect to explore in the future by changing one element in the architecture at a time, then training and evaluating the model.

Table 2. BLEU-4 Scores in chronological order starting from the original NIC implementation.

Method	Year Developed	BLEU-4 Score
Show and Tell: Neural Image Captioning [1]	2015	27.2
Caption-to-images Semantic Constructor [11]	2019	33.9
Image Captioning with Visual Relationship Attention [10]	2021	38.5
Adjustment and Augmentation	2022	31.1

Table 2 displays the BLEU-4 scores for our method along with several others that have been successful in the recent years. The key developments include semantic representation of the input image and the addition of attention [10]. While our method could not outperform these models that incorporate new and advanced technology, we show that simple and subtle changes are capable of improving performance significantly. In our case, we observed a 3.9 BLEU-4 score increase compared to the original NIC model. This is very promising since we did not change the original architecture significantly. An important takeaway from our experiments is that as new techniques emerge, they can be incorporated into existing models to improve performance without the need to completely rebuild them. In this rapidly growing field of deep learning, it is critical that we apply key innovations to maximize model functions and capabilities.

References

[1] Vinyals, Oriol, et al. "Show and tell: A neural image caption generator." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.

[2] LeCun, Y., *et al.*, "Backpropagation Applied to Handwritten Zip Code Recognition," in *Neural Computation*, vol. 1, no. 4, pp. 541-551, Dec. 1989, doi: 10.1162/neco.1989.1.4.541.

[3] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams.In Conference on Computational Natural Language Learning, 2011.

[4] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences from images. In ECCV, 2010.

[5] K. Cho, B. van Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In EMNLP, 2014.

[6] Panwar, Shwetank. NIC-2015-Pytorch. 2019. Github repository. https://github.com/pshwetank/NIC-2015-Pytorch

[7] https://albumentations.ai/

[8] Holtzman, Ari, et al. "The curious case of neural text degeneration." *arXiv preprint arXiv:1904.09751*. 2019.

[9] Mann, Ben. “How to sample from language models.” Online article. https://towardsdatascience.com/how-to-sample-from-language-models-682bceb97277. 2019.

[10] Zongjian, Z., Qiang, W., Yang, W., Fang, C.: Exploring region relationships implicitly: image captioning with visual relationship attention. Image Vis. Comput. **109**, 104146 (2021). https://doi.org/10.1016/j.imavis.2021.104146

[11] Su, J., Tang, J., Lu, Z., Han, X., Zhang, H.: A neural image captioning model with caption-to-images semantic constructor. Neurocomputing **367** (2019). https://doi.org/10.1016/j.neucom.2019.08.012

[12] https://bair.berkeley.edu/blog/2019/06/07/data_aug/

[13] Delloul, K., and S. Larabi. "Image Captioning State-of-the-Art: Is It Enough for the Guidance of Visually Impaired in an Environment?." *International Conference on Computing Systems and Applications*. Springer, Cham, 2022.