Image Captioning : CS 7643

Neel Patel*, Neelima Mehta*, Dylan Etris*, Surya Garikipati
Georgia Institute of Technology
{npatel673, nmehta39, detris3, sgarikipati}@gatech.edu

Abstract

Image captioning, the task of automatically generating natural language descriptions of images, has wide-ranging applications in areas such as accessibility and image indexing. In the realm of image captioning, conventional methods have relied on encoder CNN models to extract features from images, followed by language models decoding textual captions. However, with the rise of transformers in the field [1], we aim to compare their performance in image captioning against more traditional approaches. In this project, we conduct a comparative analysis of two prominent image captioning architectures. The first model leverages a ResNet[2] encoder with an LSTM[3] decoder and attention mechanism, while the second employs a ViT encoder coupled with a GPT-2 decoder. We rigorously evaluate these models on a standard image captioning dataset, assessing their performance through various metrics. Our report includes visualizations such as training/validation loss curves and examples of model-generated captions, providing insights into model behavior. Furthermore, the report explores the impact of architectural choices and hyperparameter tuning on image captioning results. Furthermore, we explore the interpretability of the generated captions, analyzing how well each model captures context and semantic coherence. We also investigate the efficiency of the architecture. Finally, we discuss the implications of our findings for advancing the field of image captioning and suggest potential directions for future research and development.

1. Introduction/Background/Motivation

Image captioning is a task that involves developing a model capable of processing images, and generating a description of what is occurring in those images. There are various applications for image captioning such as aiding visually impaired individuals in navigating their environment, automatically generating captions for social media posts, aiding in content moderation by auto generation of descriptions of images to flag inappropriate content, etc. It poses an interesting challenge in deep learning, since it combines different disciplines (computer vision with natural language processing) to successfully generate a caption. It also suffers from various challenges such as object hallucination, missing context, illumination conditions, contextual understanding, and referring expressions. While the scope of this work is not aimed at solving any specific problem, we wanted to use this project as means to develop an understanding of some of these issues, and experiment with the models to see what has most impact.

Image captioning is usually treated as a sequence-to-sequence problem, similar to the treatment of machine translation problems. The architecture is typically "encoder-decorder" in form, where the encoder focuses on a good representation of the objects detected in an image, while the decoder uses this representation along with language vocabulary to generate text for the images. Convolutional Neural Networks (CNN) + Recurrent Neural Networks (RNN) have been used traditionally as encoder and decoder, respectively. This has yielded significant results but not without limitations. Often, the caption generated is vague and general, and does not appropriately describe the image contents, since all the information is in a single vector. It is limited in its ability to describe deeper relations between image content.

We decided to build two different types of models - (a) CNN + LSTM, and (b) ViT + GPT2 to compare the performance of these two approaches and get a better understanding for their suitability for image captioning, and the different limitations these approaches may have.

1.1. Dataset

This project utilizes the Flickr8k dataset created with the sole purpose of advancing research in the fields of computer vision and natural language processing, particularly in the field of image captioning. The specific task was to develop algorithms that could automatically generate description captions for images, creating a link between visual content and textual understanding. The dataset was created in response to the need for standardized image-caption pairs to train and evaluate various algorithms effectively.

The dataset was created by a research group at the Cen-

ter for Language and Speech Processing at Johns Hopkins University. It was developed as part of academic research efforts. The dataset was likely funded through various sources, which would include grants from government agencies, private foundations, and academic institutions.

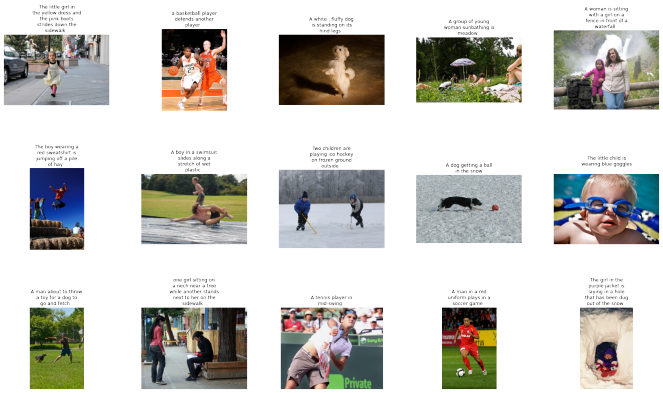

Figure 1. Data from Flickr8k

2. Approach

Our work focuses on comparing two models to discuss their differences in performance and efficiency. Both models follow an encoder/decoder architecture to compress the image into an intermediate representation, and then generate the caption.

This encoder-decoder setup is all about combining the image information with our understanding of language. The system learns to see what's in the picture and then describe it in words that feel natural. During training, we fine-tune the entire model to make the generated captions as close as possible to the 'real' captions written by people. This process teaches the model to connect the image features to the language it uses.

We wanted to compare a well-performing CNN+RNN model with a transformer-based model, and so decided not to use the vanilla architectures for the former. Instead, we used ResNet for the encoder and LSTM with attention for the decoder for the first model, while using ViT as encoder and GPT2 as decoder for the second model. LSTM, when compared to RNN, provides memory to track short or long range dependencies within limits. This improves with attention. However, our focus was on seeing if the deeper relations between contents of an image are better described with the transformer model. The primary metric for comparison was training and validation loss for both models.

2.1. Model 1: CNN + LSTM

Our first model is a CNN encoder and an LSTM decoder with self-attention. We use a pretrained ResNet [1] as our encoder. ResNet is a CNN with residual connections that increase the gradient flow during backpropagation. The parameters for the encoder are frozen so it does not get updated during training. It produces intermediate representations of the shape (batch size, feature size x feature size, number of features). The feature size is 9 and the number of features is 2048. For our decoder, we use an LSTM, which is a type of RNN that can retain long-term information in the data. Our decoder also uses self-attention, which allows the LSTM to attend to important sections of the input. Attention-based methods help emphasize the most relevant aspects of the input image, making the processing more efficient and accurate.

2.2. Model 2: ViT + GPT2

The second model also leverages pre-trained components: a ViT for understanding images and a GPT-2 transformer to turn those image features into captions. Let me break down some of how we set it up:

- **Hugging Face Tokenizer:** This tool translates our text captions into 'tokens' the model understands. We customized it so the model knows where each caption starts and ends.
- **Feature Extractor:** This is the part connected to the ViT that pulls out the important details from each image.
- **Metrics:** Cross-entropy loss helps us track how well the model's doing overall. ROUGE-2 [4] is a text-focused metric to see if the captions it generates sound natural. During training, we monitored these to fine-tune the model.
- **Image Preprocessing:** Before feeding the images in, we resized and normalized them. This makes it easier for the ViT to do its job.

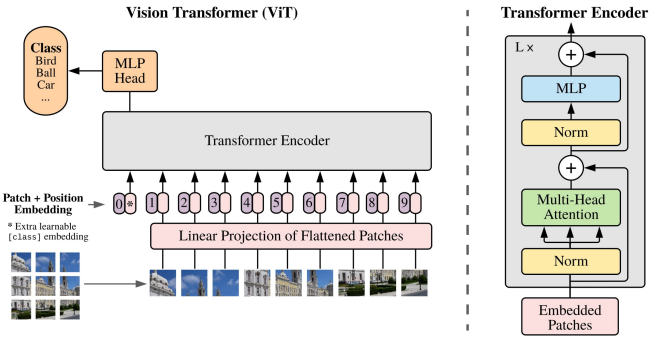

Figure 2. ViT architecture [5]

For the encoder portion of our project, we have chosen to use Google's Vision Transformer (ViT). This powerful architecture applies the transformer approach, originally designed for language tasks, to image classification.

Pre-trained on the massive ImageNet-21k dataset and later fine-tuned on ImageNet 2012, ViT offers advantages over standard convolutional neural networks (CNNs). Instead of focusing on localized features like CNNs do, ViT breaks an image into patches and treats these patches like words in a sentence. These patch ”tokens” are fed into a transformer encoder. The encoder uses layers of self-attention mechanisms and feed-forward networks to figure out how different parts of the image relate to each other, both locally and across the entire image. This attention-based approach lets ViT extract incredibly detailed visual information, making it perfect for tasks like image captioning.

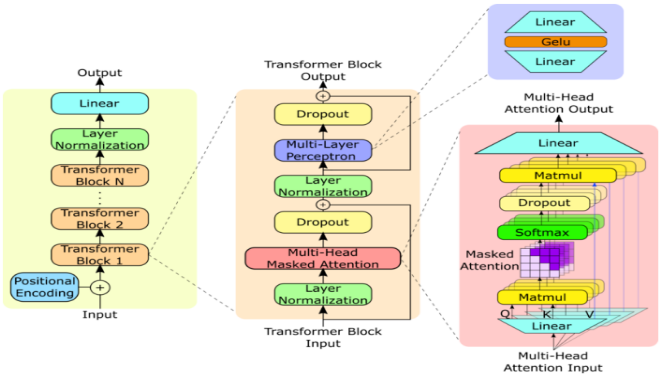

Figure 3. GPT-2 architecture [6]

For the decoder, we are using GPT-2, a powerful transformer-based language model. Each of its decoder layers carefully examines the words generated so far, helping it predict what should come next. It’s been ’trained’ on a massive amount of text, learning to produce sentences that flow naturally and make sense within their overall context. Our project uses GPT-2 to take the image features our ViT encoder extracted and turn them into captions. By understanding what’s in the image and factoring in the words it’s already generated, GPT-2 aims to produce captions that are not only accurate but sound like they could have been written by a person.

2.3. Training Method

Training method Both models were trained using stochastic gradient descent (SGD) with Cross Entropy Loss.

Cross entropy loss is calculated

$$H = -\sum p(x) \log p(x)$$

We split the training and validation datasets 80/20 to help keep our models from overfitting. We can gain a better understanding of the model’s potential for generalization to new images by tracking its performance on this unseen validation set.

Experiments are run on each model to see how each behaves when different hyperparameters are run. The performance of the fine-tuned model is then compared.

2.4. Challenges

Training a transformer model on a subset of 500 examples from our dataset was initially a strategy to reduce training time, which it did effectively. However, this approach also presented us with new challenges and advantages that were not initially expected. One notable disadvantage was the increased risk of over-fitting due to the limited diversity and representation in the smaller subset. This overfitting became evidence as the models showed high accuracy on subset but struggled when exposed to new, unseen data. On the bright side, this experiment provided us with valuable insights into how the models behaved with varying training data sized. It highlighted the importance of dataset size in model generalization and performance, showcasing the trade-offs between training efficiency and model robustness.

3. Experiments and Results

3.1. CNN+LSTM

The baseline hyperparameters for the CNN+LSTM model are shown in table 1. Each experiment varies one of the listed hyperparameters, and learning curves are provided to show the effect. We saw overfitting throughout all experiments, which suggests that this model may be powerful enough to memorize the training dataset.

Hyperparameter	Value
Decoder dimension	512
Embedding size	300
Attention dimension	256
Dropout probability	0.3
Training dataset size	3000

Table 1. Default hyperparameters for the CNN+LSTM model

3.1.1 Varying training dataset size

The effect of training on 500 data examples is shown below. This is the version with lowest data examples. The lowest validation loss for this run was around 4.0. The effect of training on 1000 data examples is also shown. This did not show significant improvement either. We saw some improvement with 3000 data examples, though it was combined with changes in other hyperparameters. Validation loss was around 3.7

3.1.2 Varying learning rate

The learning curve for learning rate of 3e-4, 3e-3 and 8e-4 are shown in figure 4. A higher learning rate resulted in greater overfitting.

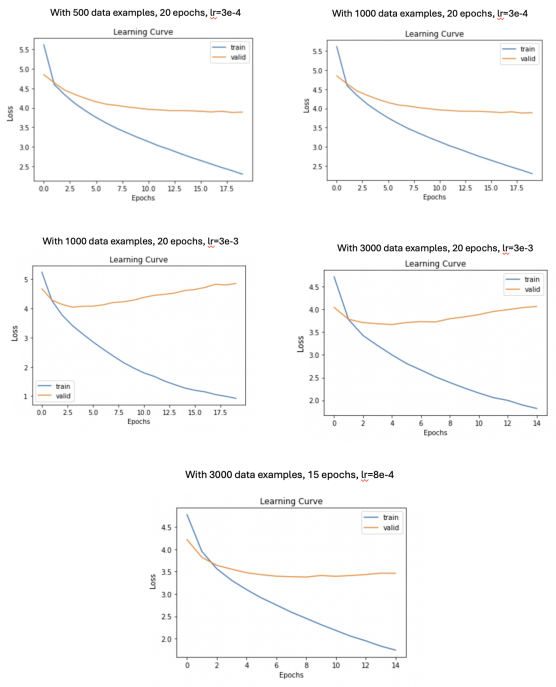

Figure 4. CNN + LSTM variations in hyperparameters

3.1.3 Varying Dropout

Table 2 shows the effect of varying dropout probability, training on 3000 examples for 10 epochs with a 80/20 test/validation split. The other hyperparameters are the same as in table 1. The learning curve for the best dropout is shown in figure 5. This gave us our best result for this model - validation loss of around 3.4. We wanted to try dropout probability as a regularization method to reduce the overfitting, so we did expect improvement. However, it was an unexpected result that a low dropout probability performed so well, because the hypothesis was a higher dropout probability would reduce overfit. On further research, we realized this was due to the dataset being relatively small (only 3000 examples), that even a low dropout probability quickly reduces overfitting.

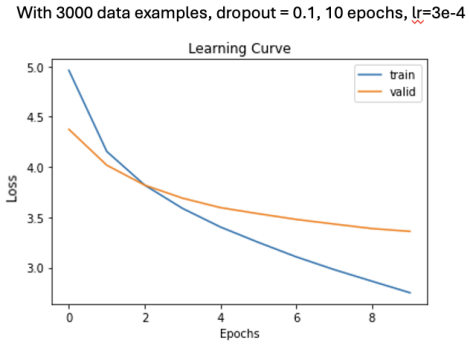

Figure 5. Best Dropout Model for CNN+LSTM

Dropout Probability	**0.1**	0.2	0.3	0.4	0.5
Loss	**3.40**	3.42	3.46	3.50	3.53

Table 2. CNN + LSTM Loss for different dropout probabilities. The best dropout probability and loss are in bold.

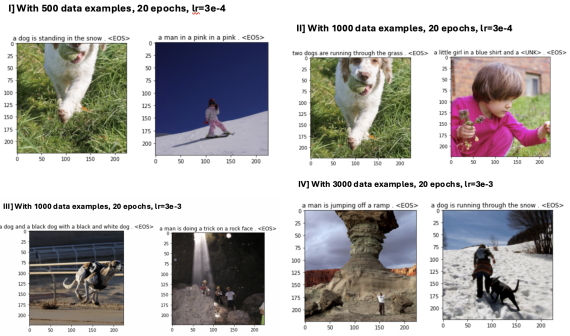

Figure 6. CNN + LSTM image captioning examples by variations

3.2. ViT+GPT2

The notebook we used as a baseline was already tuned very well, and it was hard to see drastic improvements. We ran a few experiments to address the problems of computation costs and over-fitting.

3.2.1 Hyperparameter Tuning

The initial model included the hyperparameters as described in figure 8. The training loss and validation loses the base model and the tuned model are given in figure 7

Model	**Training Loss**	**Validation Loss**
Initial	1.5958	2.48495
Tuned	1.38594	2.32748

Figure 7. Hyperparameters for the Baseline Transformer Model

The first hyperparameter we trained was the learning rate. Learning rate was decreased to 5e-3. This lower

learning rate helped the model to converge more gradually, ensuring a smooth navigation through the optimization landscape rendering the model more effective without overshooting or getting stuck in a local minima. The second hyperparameter we tuned was to increase the training batch size from 8 to 16, leading to more stable gradients and smoother optimization, resulting in improved generalization and reduced over-fitting. By processing more data in each training iteration, the model was able to learn more efficiently and effectively while capturing complex patterns in the data, leading to a better training and validation losses. The last hyperparameter we tuned was the weight decay. The reason behind this was to prevent excessive regularization during training. By decreasing weight decay to 5e-3, we ensured that the model was able to main more flexibility and adaptability in learning complex patterns in the data, ultimately contributing to better generalization and lower validation loss.

In summary, the fine-tuning of these parameters resulted in much better performance by providing a More controlled and stable learning environment, allowing the model to employ efficient data processing and preventing excessive regularization. Each parameter adjustment contributed to the improvements, collectively leading to the observed reduction in training and validation losses and overall enhanced model performance.

Parameter	**Inital**	**New**
Learning Rate	1e-2	5e-3
Training Batch Size	8	16
Weight Decay	1e-2	5e-3

Figure 8. Hyperparameters for the Baseline Transformer Model

3.2.2 Reducing Data

Each epoch, when training on the entire dataset, required over 3 hours. Completing 5 epochs took approximately 15 hours! To tackle the challenge of computation costs, our goal was to diminish the dataset's size to observe if the model's performance plateaued after a specific data threshold.

Table 3. Data Reduction

	500 ex.	2000 ex.	4000 ex.	8000 ex.
Train. Loss	2.03	0.74	0.64	0.59
Val. Loss	2.84	3.42	3.41	3.43
R2 Prec.	0.03	0.03	0.03	0.03
R2 Recall	0.24	0.30	0.30	0.30
R2 F-meas.	0.05	0.05	0.05	0.05
Time(min)	20	68	135	270

The table showcases the impact of dataset reduction on training and validation metrics alongside corresponding

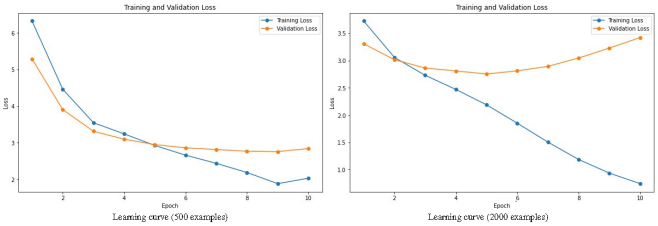

Figure 9. Learning curve (500 & 2000 examples)

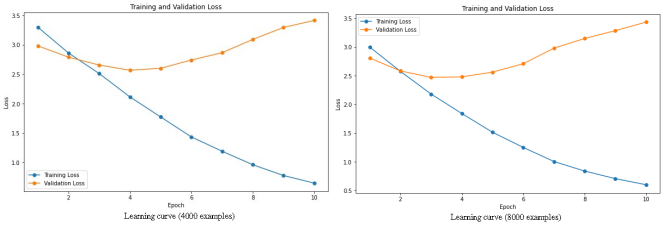

Figure 10. Learning curve (4000 & 8000 examples)

time requirements. As the dataset size decreases, training loss reduces while validation loss increases, indicating potential overfitting on smaller datasets. This aligns with machine learning theory, where smaller datasets allow for better fitting but might lead to poorer generalization. Stability in precision, recall, and F-measure across dataset sizes suggests less sensitivity to changes. Increased computation time with larger datasets is expected. The methodology of gradual dataset reduction provides insights into the trade-off between model performance and computational efficiency, guiding future dataset curation and training decisions.

3.2.3 Early Stopping

It's evident that the Transformers notebook is susceptible to over-fitting. In each experiment, the training loss consistently decreases, while the validation loss begins to increase past a certain threshold. This clear divergence indicates over-fitting. To address this issue, we implemented early stopping. If the validation loss fails to improve over 3 iterations, training halts. Given the large dataset size, we opted for step-based monitoring rather than epochs. The 'logging_state' was set to 1000, meaning if the validation loss doesn't increase over 3000 steps, training ceases. The model underwent training for 10 epochs, although some stopped training earlier. We conducted a comparison between the baseline and two additional models with hyperparameter tuning.

	Training Loss	Validation Loss	Stopped Early
Baseline	1.56	2.28	Yes
Model 1	1.89	2.36	No
Model 2	1.66	2.34	Yes

The table illustrates the training and validation losses,

alongside early stopping outcomes, for the baseline model and its hyperparameter-tuned iterations, Model 1 and Model 2. The baseline model achieves the lowest training loss, indicative of its ability to fit the training data closely. However, this comes at the cost of a higher validation loss, signaling over-fitting. Model 1, the result of hyperparameter tuning from the baseline, shows a higher training loss but similar over-fitting tendencies. Conversely, Model 2, also a product of hyperparameter tuning, demonstrates a slightly higher training loss than the baseline but showcases improved generalization, reflected in its lower validation loss. This underlines the efficacy of early stopping strategies, as implemented in Model 2, in mitigating over-fitting. The utilization of step-based monitoring acknowledges the dynamic nature of training, ensuring adaptability in termination criteria.

3.3. Results

Our experiments with both models yielded some expected and some unexpected results. Transformers are powerful models that have gained much popularity recently, and are particularly suited for NLP-based applications. Their ability to conduct operations in parallel, by using positional embeddings for longer-term context, and their inherent self-attention mechanism puts them at an advantage. Thus, even though our CNN+LSTM model had multiple sophisticated modifications (ResNet with dropout probability, and LSTM with attention), it did not exceed the performance of the transformer-based ViT+ GPT model. This was expected, though it was surprising to see that the worst performance of the transformer-based model (3.4) was similar to the best performance we got with CNN+LSTM model. The best result we got with ViT + GPT was 2.23 which was a significant improvement.

4. Conclusion & Future Work

In conclusion, our project embarked on an in-depth analysis of image captioning, comparing and contrasting two of the most prominent architectures: CNN + LSTM and ViT + GPT2. Through meticulous experimentation and thorough analysis, we were able to acquire some valuable insights into the strengths and limitations present in each approach.

The CNN+LSTM model underwent extensive examination, with a focus on fine-tuning hyperparameters such as dropout probability, and learning rate. This effort aimed to find an optimal balance between training efficiency and model generalization, a delicate balance in deep learning. Despite facing challenges like over-fitting due to the model's complexity, our strategic adjustments showed promising results, displaying the model's adaptability and potential for detailed image captioning.

On the other hand, the ViT + GPT2 architecture demonstrated remarkable ability in capturing complex relationships between image features and their textual representations. However the computational demands of this advanced model posed significant challenges, requiring innovative strategies such as data reduction and early stopping techniques to mitigate over-fitting and improve training efficiency.

Looking ahead, out project's findings shed light on potential research areas in the field of image captioning. It is evident that there is a good amount of room for optimizing transformer-based architectures, exploring new attention mechanisms, and leveraging large datasets to enhance model performance and generalization. Additionally, the pursuit of interpretability and context-aware caption generation presents promising deeper insights into the relationship between computer vision and natural language processing.

Ultimately, this project not only deepened our understanding of image captioning methodologies but also highlighted the dynamic interplay of challenges and opportunities in computer vision and natural language processing. As we continue to explore this fascinating field, we anticipate ongoing advancements that will redefine AI-driven image understanding interpretation.

5. Work Division

5.1. Dylan

- Set up the initial training pipeline for CNN+LSTM
- Ran the dropout probability experiment
- Contributed to the Approach and Experiment sections

5.2. Surya

- Set up the initial training pipeline for ViT+GPT2
- Ran the early stopping and reduced data experiments for the transformer notebook
- Contributed to the Abstract, Approach, and Experiment sections

5.3. Neelima

- Ran experiments for CNN+LSTM model, and some for ViT+GPT2
- Contributed to the Introduction/Motivation, Approach, Experiments and Results sections

5.4. Neel

- Ran ViT + GPT2 Model
- Contributed to the Abstract, Approach, Experiment Results, and Conclusion Sections

References

[1] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).

[2] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[3] Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. "Sequence to sequence learning with neural networks." Advances in neural information processing systems 27 (2014).

[4] Lin, Chin-Yew. "ROUGE: A Package for Automatic Evaluation of Summaries." Annual Meeting of the Association for Computational Linguistics (2004).

[5] Image taken from the website: https://huggingface.co/docs/transformers/en/model_doc/vit

[6] Yang, Steve Ali, Zulfikhar Wong, Bryan. (2023). FLUID-GPT (Fast Learning to Understand and Investigate Dynamics with a Generative Pre-Trained Transformer): Efficient Predictions of Particle Trajectories and Erosion. 10.26434/chemrxiv-2023-ppk9s.

[7] He, K., Zhang, X., Ren, S., Sun, J.(2016) Deep residual learning for image recognition,*In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778)*,

[8] Gandhi T., Pourreza H., Mahyar H., (2023). Deep Learning Approaches on Image Captioning: A Review. *In ACM Computing Surveys, Bolume 56, Issue 3, (pp.139)*,

6. Appendix

Table 5. Model Hyperparameters

	Model 1	**Model 2**
Train Batch Size	16	32
Val Batch Size	16	32
LR	1e-5	2e-5
Weight Decay	0.001	0.005
Top-k	50	20
Top-p	0.85	0.75
Length Penalty	1.5	1.0
Beam Num.	6	8
Warmup Steps	300	300

Table 6. Image Captions

	Caption Image 1	**Caption Image 2**
Baseline	A black dog and a brown dog run side by side on the street...	A little girl in a pink dress is climbing up a wooden staircase in a wooded area.. She is about to enter a
Model 1	A black and white dog is running with a black dog in its path.. The blac	A little girl in a red coat is standing in front of a wooden structure. She is looking through a window. An
Model 2	A black and white dog is running on the sidewalk next to a white dog. Th	A little girl in a pink dress sits on a wooden bench in front of a wooden house... a little boy in a red dr

Figure 11. Image 1 & Image 2