Image Captioning Multi-Modal Model Vectorization and Hyperparameter Configuration Evaluation

Derek Plemons

MSDS 458: Artificial Intelligence and Deep Learning

Dr. Syamala Srinivasan

November 28, 2024

Abstract

This research study evaluates the effectiveness of multiple hyperparameter configurations and vectorization approaches in multi-modal deep neural networks (DNN). The main task of the DNN is to caption images using PyTorch, Long Short-Term Memory (LSTM), a pre-trained BLIP model and the Flickr30k dataset. This study systematically compares different strategies of optimization on local and cloud computing machines which include Adam, AdamW, RMSprop, SGD, and Adagrad optimizers. These are then combined with different learning rate schedulers and loss functions to create a total of 45 unique experiments per vectorization method. The total number of experiments comes out to 180 with a quarter of the experiments being conducted in the cloud environment. We then use a pre-trained BLIP model to generate captions for comparison on a sample of 10 images to compare to the PyTorch model performance metrics.

The experimentation framework that is implemented utilizes a grid search across hyperparameters. Performance is evaluated using multiple metrics such as BLEU, ROUGE, and METEOR scores. The training was done on the Flickr30k dataset which contains 31,000 images paired with five annotations created by humans. The optimal configuration of vectorization methods and hyperparameters was the CLIP vectorization method with standard Cross-Entropy Loss combined with Adagrad optimizer and the exponential Learning Rate Scheduler. The Bootstrapping Language-Image Pre-training (BLIP) image captioning model had a higher performance than the PyTorch/LSTM multi-modal models which was likely due to training on a small sample size as a result of computational resource limitations.

Introduction

There are many utilities for integrating computer vision (CV) and natural language processing (NLP) across industries. Image captioning is the task of generating natural language descriptions of images. Image captioning has become a useful application in making visual context accessible to visually impaired users, enhancing search engine results, and improving content management systems in social media platforms. Recent advances in deep learning have drastically changed the landscape of what is possible to accomplish. Separately, CV and NLP present significant challenges but combining these modalities presents additional challenges. Image captioning

systems must be able to recognize objects, attributes, and relationships within the image. They must take these observations and be able to express them in a manner which is easily understandable and accurate. This task requires a sophisticated neural architecture that can bridge the semantic gap between textual and visual representations. Bridging the gap can present challenges. One of the primary challenges lies in how to optimize these multi-modal architectures as their performance is highly reflective of the careful selection of hyperparameters and vectorization methods. The selection of optimization algorithms, learning rate schedulers, and loss functions can and does significantly affect the model performance in terms of captions generated and computational efficiency.

The current state-of-the-art approach to solving multi-modal problems utilizes transformer based architectures. These have been shown to have high success rates in both CV and NLP tasks. Transformer model performance is highly sensitive to hyperparameter configuration and there is a lack of systematic research in optimal hyperparameter selection for multi-model tasks. The gap in knowledge around hyperparameter selection leads to reduced model performance and training processes which are inefficient.

The implications of this research could be potentially significant given the number of people globally that are visually impaired. According to the World Health Organization, around 285 million people worldwide are visually impaired. An effective image captioning system could significantly improve their ability to interact with their environment and visual content on the internet. The growing volume of such visual content on the internet, social media, and news programs highlights the need for automated systems that can generate accurate and natural descriptions for content discovery and search engine optimization.

This research study aims to systematically evaluate multiple hyperparameter configurations and optimization strategies for multi-modal deep neural networks. Through the course of this study we will compare the effectiveness of multiple optimization algorithms such as Adam, AdamW, RMSprop, SGD, and Adagard for image captioning. We will also assess the impact of different learning rate schedulers on model convergence and caption quality. We will evaluate loss functions and variations of Cross-Entropy Loss without and without label smoothing. We will also analyze the relationship between different hyperparameter combinations and their

performance using BLEU, ROUGE, and METEOR performance metrics. Finally, we will compare the custom multi-modal models performance against pre-trained BLIP models.

Through this research we aim to produce empirical evidence and practical guidelines for those seeking to build and implement image captioning systems. The findings of this study will contribute to the academic understanding of multi-modal deep learning systems and address the gap in literature providing systematic evaluation of hyperparameter configurations. More specifically, how these configurations affect caption generation quality.

Literature review

The first research studies into image captioning began with approaches that were based on templates. However, the field of image captioning grew exponentially with the computer processing advancements and deep neural networks. Vinyals et al. (2015) created a "Show and Tell" model that combined convolutional neural networks (CNN) for image processing and Long Short-Term Memory (LSTM) networks for text generations. This established the foundation for the modern image captioning systems we see today.

The "Bottom-Up and Top-Down Attention" approach was pioneered by Anderson et al. (2018). This system utilized visual attention mechanisms to identify salient image features during caption generation. This study demonstrated the significance of attention mechanisms in caption quality and relevance. In doing so they were able to achieve excellent results on the MSCOCO dataset.

Xu et. al (2020) conducted research into hyperparameter optimization strategies for image captioning models. Their work examined the impact of learning rate schedulers and loss functions on caption quality. They found that adaptive learning rate methods such as AdamW combined with label smoothing produced the highest accuracy.

The "Oscar" model was introduced by Li et al. (2020) and is a transformer-based architecture that unifies vision and language pre-training. Their research found that caption generation accuracy is improved by object-attribute detection. Zhang et al. (2021) built on this research and

developed “VinVL” which enhances visual features with greater scale pre-training and object detection.

Salesforce Research (2022) introduced the model Bootstrapping Language-Image Pre-training (BLIP). This model uses an image-text pair filtering mechanism and multi-task learning approach. BLIP demonstrated excellent performance across vision and language tasks which include image captioning.

Vedantam et al. (2015) introduced CIDEr that was specifically designed for image captioning assessment. This study elaborates on the limitations of traditional metrics such as BLEU and ROUGE for image captioning tasks. Yi et al. (2020) who performed a comprehensive analysis of evaluation metrics found that a combination of metrics is more reliable when assessing caption quality.

Wang et al. (2020) conducted a study of hyperparameter optimization for image captioning models. Their findings indicate that the careful selection of optimizers and learning rate schedulers impact training stability and model performance. Their study focused on the MSCOCO dataset. The current body of research related to this current study highlights gaps in the hyperparameter configurations for multi-modal networks using the Flickr30k dataset.

Methods

The research methods employed in this study use a deep learning approach to develop a multi-modal image captioning system using the Flickr30k dataset. The methodology includes comprehensive data preparation, data analysis, model architecture design, and comprehensive evaluation procedures.

To prepare the dataset we first extensively preprocess the image captions. This process began with text cleaning operations to remove symbols, lowercasing text, punctuation, and normalization of special characters. A new column was created for this cleaned text. Then another column was created with captions where the stop words were removed using the NLTK english stop words list to reduce noise in the textual analysis.

For exploratory data analysis we primarily evaluated the characteristics of the captions. We did this by calculating the lexical diversity using the Type-Token Ratio (TTR) metric which provides insight into vocabulary richness in the entire corpus of captions. Name Entity Recognition (NER) was performed using spaCy en_core_web_sm model to identify and categorize entities within captions. This enabled us to understand the distribution of different entity types such as people, places, and organizations. We analyzed the caption length distribution to identify potential outliers and better understand the syntactic complexity of the captions. Then we calculated caption similarity for a subset of images using cosine similarity on TF-IDF vectors to assess annotation consistency.

For the multi-modal model architecture we used three vectorization approaches: BERT-base-uncased, RoBERTa-base, and CLIP-ViT-base-patch32. Due to computational constraints we initially trained on a random selection of images and captions which comprised 0.001% of the full dataset. A random sample was selected to ensure a balanced representation across different caption types. The foundation of the image processing pipeline utilized a PyTorch convolutional neural network architecture. The final classification layer was removed and adaptive average pooling was implemented to ensure consistent output of dimensionality of 2048 image features regardless of input image size. Then a linear projection layer was created to reduce the dimension to 512 to match the LSTM hidden state size. The CNN's early layers were frozen to leverage pretrained features while allowing fine-tuning of the final convolutional block.

The multi–modal model component that generates captions utilized a two-layer LSTM network with a hidden state of 512 units. The word embedding layer was initialized with a GloVe 6B 300 dimensional vector which mapped input tokens to dense representations. The forward pass concatenated images features with embedded captions tokens allows the LSTM to learn joint visual-textual representations. We used teacher forcing during training with a ratio of 0.5 and gradually decreased this value over training epochs to improve model robustness.

The training involved a systematic hyperparameter grid search across multiple configurations. We evaluated three loss functions: Standard Cross Entropy Loss, Cross Entropy with label smoothing, and Negative Log Likelihood Loss. All of the loss functions were configured to ignore padding tokens. Five optimizer configurations were tested: Adam, AdamW, RMSprop, SGD, and Adagrad. Three learning rate scheduling strategies were also implemented: StepLR,

ExponentialLR, and CosineAnnealingLR. The grid search evaluated all possible combinations of configurations for 20 epochs with an early stopping patience of 5 epochs based on validation loss. The total number of combinations per vectorization approach is 45. With all three vectorization approaches there were a total of 135 hyperparameter combinations that were tested on my local machine. Additionally, 45 experiments were performed using Google Cloud using the CLIP vectorization method on 0.01% of the data.

To evaluate the results of caption quality we used multiple NLP metrics. BLEU scores were calculated using NLTKs implementation with smoothing method 7 which considers n-grams up to n=4. ROUGE metrics: ROUGE-1, ROUGE-2, ROUGE-L, were calculated using the rouge-score library. METEOR scores were also calculated using the nltk.translate.meteor_score library with exact, porter stem, and WordNet synonym matching. The multi-modal model was implemented using my local laptop with a nvidia geforce GPU and a cloud single NVIDIA A100 GPU with 40GB memory.

For a baseline comparison, we used a pre-trained BLIP model created by Salesforce for image captioning to compare our experiments against. This was run on my local machine. We selected the first 10 images in the dataset and made the BLIP model generate captions. We then performed a similar analysis where we used the BLEU, ROUGE, and METEOR scoring methods to evaluate how accurate the generated captions were to the human annotated captions.

Results

The exploratory analysis of the Flickr30k dataset revealed a structure composition of 31,783 unique images paired with 158,914 unique captions. There is a consistent annotation density of five captions per image. Upon examining the caption lengths histogram plot (Figure 3) we can see a clear right-skewed distribution. The majority of the captions contained between 10 and 20 words. The distribution has a peak at approximately 12 words per caption with a sharp decline in frequency beyond 20 words. The scarcity of captions with greater than 40 words indicates an upper bound for captions in this dataset.

The analysis of the linguistic composition of the captions was carried out through Part-of-Speech (POS) tagging. This revealed that nouns were the highest frequency with 454,107 occurrences, followed by adjectives with 189,903 instances, and plural nouns with 139,877 instances. The high frequency of present participles with 138,072 instances indicates that there is a strong element of describing the images captions which is what we expect to see in a dataset of captions that are supposed to be describing the images. A Type-Token Ratio (TTR) of 0.02 was found and indicates that there is a significant amount of vocabulary repetition across the captions. This high amount of repetition makes sense given the nature of the captions being visual descriptions of the images where common objects recur across different images.

The NER analysis uncovered patterns related to the types of references in the dataset captions. There are 38,260 numerical references in the captions which were the most frequent entity type. Nationality, religious, and political group references appeared 3,173 times. Temporal references were not featured significantly in the captions dataset with only 209 date entities and 86 time entities. This makes sense as the captions are describing the images and it wouldn't make as much sense to include temporal references. Person names also appeared infrequently with only 59 occurrences. This indicates that the captions generally favored descriptive content over specific identification. Geographic and location entities were also sparse with 131 and 5 occurrences. This suggests that the captions have a focus on scene and action instead of spatial contextualization.

The caption similarity analysis of the first ten images in the dataset revealed significant variation in descriptive consistency. The highest similarity score was 0.701 percent and the lowest with 0.219. The overall average similarity score was 0.48 across all 10 images. This suggests that while captions for the same image share common descriptive elements, they exhibit significant linguistic variation in their expression. This also suggests that the captions are varied enough to provide multiple possible descriptions of an image while still capturing the essence of the image. The moderate level of similarity may indicate that the individuals who annotated the images provided consistent and relevant descriptions of the images. Three images in particular have a similarity score above 0.6. This indicates that the captions were very consistent for those images.

The above exploratory data analysis findings have significant implications for model development and evaluation. The caption length distribution may indicate that implementing a

maximum sequence length of 40 words would be able to cover the majority of cases without a loss of information. The low lexical diversity may indicate the potential advantages for word embedding that capture semantic similarities for frequently recurring terms. The high frequency of present participles may indicate the importance of designing a model architecture that can capture and represent action-based descriptions. Finally, the patterns in caption similarity suggest that evaluation metrics would be best designed to score alternative descriptions of the same visual content. Evaluating the generated captions in terms of n-gram comparisons may not be the best approach given the variation in annotated captions.

The hyperparameter grid search experiments (Figure 5) across different vectorization methods and compute environments yielded significant insights about model performance and optimization strategies for image captioning. The output in Figure 5 contains the Top 5 best performing experiments for each method which comes out to 20 experiments. The remaining 160 experiments, their configuration and scoring output can be found in the attached excel file. By comparing local machine, cloud computing, and pre-trained models we were able to derive valuable insights into methods for generating image captions.

The local machine experiments that were conducted on 0.001% of the dataset across three difference vectorization methods revealed interesting insights. The total time of training and hyperparameter selection took 3.5 hours on my local machine. The three vectorization methods that were used are: BERT-base-uncased, RoBERTa-base, and CLIP-ViT-base-patch32. The CLIP vectorization experiments demonstrated the most promising results of all the local experiments. CLIP's architecture was designed for zero-shot transfer learning. This allows it to generalize well even with minimal fine-tuning and is reflected in the performance stability across different hyperparameter configurations. CLIP's visual encoder was trained to extract features relevant to textual descriptions. This is a fundamental difference from BERT and RoBERTa which must learn image to text relationships from scratch during fine-tuning.

The best configuration for the local machine experiments was a criterion = CrossEntropyLoss, optimizer = Adagrad and scheduler = StepLR and it achieved a BLEU score of 8.895936112940164e-232, ROUGE score of 0.194 and METEOR score of 0.069. The scoring output suggests that CLIP's multi-modal pre-training provides advantages over the other vectorization methods even with the limited training data. The Adagrad strong performance may

be attributed to its adaptive learning approach. While other learning rate optimizers may be fixed, Adagrad adapts the learning rate for each parameter based on historical gradients. This is beneficial for image captioning because different components of the model may require different learning rates. Adagrad automatically adjusts for different learning rates by scaling learning rates inversely proportional to the square root of accumulated squared gradients. Parameters that receive large or frequent updates would get smaller learning rates while the inverse would get larger learning rates. This adaptive behavior helps to balance the learning between visual and textual components.

BERT-base-uncased showed moderate performance with the best configuration achieving a ROUGE of 0.161 and METEOR score of 0.038. The model appeared to struggle with maintaining consistent performance across different scheduler configurations. This suggests a higher sensitivity to learning rate adjustments. While BERT itself is a powerful model, its architecture was designed and pre-trained for purely text understanding tasks. When this vectorization method is applied to image captioning, it must learn to process visual information without the benefit of prior pre-training on visual features. This explains the inconsistent performance across different scheduler configurations.

The RoBERTa-base vectorization best configuration achieved a ROUGE score of 0.157 and METEOR score of 0.091. While the ROUGE score was lower than both CLIP and BERT-base-uncased, it achieved a competitive METEOR score that was higher than both other vectorization methods. This indicates that the RoBERTa-base vectorization method may have better semantic alignment in its generated captions. The METEOR score is significant because it evaluates the semantic similarity by taking into consideration the synonyms, stemming, and paraphrases. On the other hand, ROUGE focuses more on n-gram overlap. This may explain why RoBERTa could achieve a lower ROUGE score while achieving a higher METEOR score. RoBERTa's robust pre-training that includes larger batch sizes and dynamic masking appear to provide stronger semantic understanding that transfers well to caption generation. The model's pre-training that helps generate captions capture the underlying meaning more effectively than BERT.

The experiments on the A100 cloud compute environment using 0.01% of the dataset with CLIP vectorization demonstrated significantly higher performance metrics as compared to local

machine experiments. This makes sense as the training was conducted on more data. The most successful configuration utilized Adagrad optimizer with StepLR scheduler and CrossEntropyLoss. This configuration achieved a ROUGE score of 0.199 and METEOR score of 0.128. This configuration significantly outperformed the other combinations with the second best configuration being Adagrad with CosineAnnealingLR and a ROUGE score of 0.193 and METEOR score of 0.129. The performance from Adagrad is possibly due to its adaptive learning rate approach. This appears to be an ideal approach for the task of image captioning when working with larger data samples. It is also interesting to note that the best configuration of the A100 experiments are also the same best configuration of the local machine experiments. This indicates that configuration is ideal for captioning tasks on either compute environment.

The BLIP image captioning experiments (Figure 6) across 10 test images reveal several patterns and insights. Like in the PyTorch models, the BLEU scores were very low except for some outliers. Three of the generated BLIP captions had a BLEU score of 1, 0.51 and 0.43. These are the highest BLEU scores we have seen in this study and likely indicative of the pre-trained BLIP models accuracy. The ROUGE-1 score averaged around 0.316 and the METEOR scores average around 0.184. Almost all of the scores were higher than the Pytorch model. The highest Rouge Score was 0.51 for the “a man on a ladder” caption. The highest METEOR score was 0.337 for “a man sitting on a chair.” These caption metrics performed consistently well across all metrics which indicates that the BLIP model was accurately able to capture the essential elements of their images.

There were several observations that could be made across all experiments. The impact of the selected optimizer was significant. The Adagrad optimizer outperformed all other optimizers across all vectorization and compute environments. On the other hand, the SGD optimizer consistently showed the poorest performance. This indicates that the SGD optimizer fails to learn meaningful representations. The StepLR scheduler generally provided the most stable learning progression. Although, CosineAnnealingLR showed a comparable performance in several experiments. The Exponential LR scheduler often showed poor performance which indicates that the aggressive learning rate decay may be problematic for image captioning. The loss functions CrossEntropyLoss and NLLLoss showed similar performance. CrossEntropyLoss generally achieved better performance results. This may indicate that the direct probability interpretation in CrossEntropyLoss may be more suitable for caption generation. Additionally, the order of

magnitude difference in dataset size between local and cloud computing environments (0.001% and .01%) appeared to have significant impacts on model performance. The A100 experiments consistently outperformed the local machine experiments with higher performance scores. This makes sense as the models benefit from having additional training data even when training on a small fraction of the entire dataset.

The extremely low BLEU scores across all experiments may indicate a limitation of this score in being able to evaluate image captioning tasks. BLEU requires exact n-gram matches which may not capture the semantic similarity of generated captions. Given that there are 5 different captions per image and there is significant variation among those captions as we saw in the exploratory data analysis, it makes sense that the BLEU score is not sufficient for evaluating image captioning tasks. The ROUGE and METEOR scores were a lot more promising. This suggests that those metrics would be more appropriate for evaluating caption generation tasks as they better account for semantic similarity and word order flexibility. The pre-trained BLIP model demonstrated exceptional performance that far exceeded the PyTorch models. This was likely due to the PyTorch models being trained on a small sample of the entire dataset.

Conclusions

This study evaluating hyperparameter configurations and vectorization approaches in multi-modal deep neural networks for image captioning has revealed several significant insights with implications for research and practical applications. This study was able to demonstrate that the vectorization method and hyperparameter configuration significantly impacts the performance of image captioning multi-modal models. The CLIP vectorization method outperformed both BERT-base-uncased and RoBERTa-base vectorization methods. This is likely due to CLIP's design for zero-shot transfer learning and pre-training on textual-visual relationships. The advantage of CLIP is apparent in the higher model performance across different hyperparameter configurations.

In regard to optimization strategies, Adagrad proved to be the best optimizer in all tested environments and configurations. The adaptive learning rate of Adagrad was likely the cause of its superior performance. This is because Adagrad balances the complex learning dynamics

between textual and visual features. The benefits of the Adagrad optimizer were apparent when combined with the StepLR and Cross-Entropy Loss schedulers. The combination was able to achieve the highest performance metrics in both local and cloud computing environments.

In this study we were also able to provide insights into which evaluation metrics were most useful for image captioning tasks. The very low BLEU scores in all experiments in contrast with ROUGE and METEOR score may indicate that the traditional n-gram metrics may not be useful for evaluating image captioning. This is in alignment with our exploratory data analysis findings which showed significant differences in image captions for the same image.

The pre-trained BLIP model demonstrated superior performance when compared to the PyTorch models in all performance metrics. This was likely due to the BLIP model being trained on more data while the PyTorch models were trained with a small sample size due to computation and resource limitations. However, the BLIP model still had difficulty generating captions with a high score for BLEU, ROUGE, or METEOR. This suggests that image captioning is a challenging problem that still needs additional development in research to generate captions that appear to be annotated by humans.

This study demonstrated practical recommendations for implementing image captioning systems. First, CLIP based vectorization methods over traditional models work well with limited computational resources or training data. Secondly, the Adagrad optimizer with StepLR or CrossEntropyLoss may be the most effective combination across local or cloud computing environments. Thirdly, ROUGE and METEOR scores are better for image captioning evaluation as they better capture semantic similarity and linguistic variation in image captioning tasks. Lastly, local machines are not sufficient to train a multi-modal image captioning model due to computational limitations. Instead, cloud computing tools should be utilized where accurate image captioning is needed.

Future research should focus on further evaluating hybrid performance metrics to better account for semantic similarity while maintaining sensitivity to descriptive accuracy. Additional studies would also benefit from utilizing 100% of the dataset to increase model accuracy. Given the limitations in computing time and resources, it was not feasible in this study to utilize the entire dataset for training. It may also be useful to investigate adaptive hyperparameter selection methods based on available computational resources and dataset characteristics.

Citations

1. 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 (pp. 6077-6086). https://doi.org/10.1109/CVPR.2018.00636
2. Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., Choi, Y., & Gao, J. (2020). Oscar: Object-semantics aligned pre-training for vision-language tasks. In European Conference on Computer Vision (pp. 121-137). Springer, Cham. https://doi.org/10.1007/978-3-030-58577-8_8
3. Vedantam, R., Lawrence Zitnick, C., & Parikh, D. (2015). CIDEr: Consensus-based image description evaluation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4566-4575). https://doi.org/10.1109/CVPR.2015.7299087
4. 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 (pp. 3156-3164). https://doi.org/10.1109/CVPR.2015.7298935
5. Wang, Y., Wang, H., Yang, Y., & Yang, X. (2022). On the importance of hyperparameter optimization for image captioning. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(11), 7862-7877. https://doi.org/10.1109/TPAMI.2022.3176647
6. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., & Bengio, Y. (2020). Optimization strategies for neural image caption generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 8721-8730).
7. Yi, Y., Deng, H., & Hu, J. (2020). Improving image captioning evaluation metrics using transformer-based models. In Findings of the Association for Computational Linguistics (pp. 4125-4131). Association for Computational Linguistics.
8. Zhang, P., Li, X., Hu, X., Yang, J., Zhang, L., Wang, L., Choi, Y., & Gao, J. (2021). VinVL: Revisiting visual representations in vision-language models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 5579-5588). https://doi.org/10.1109/CVPR46437.2021.00553

Appendix

Figure 1

Figure 2

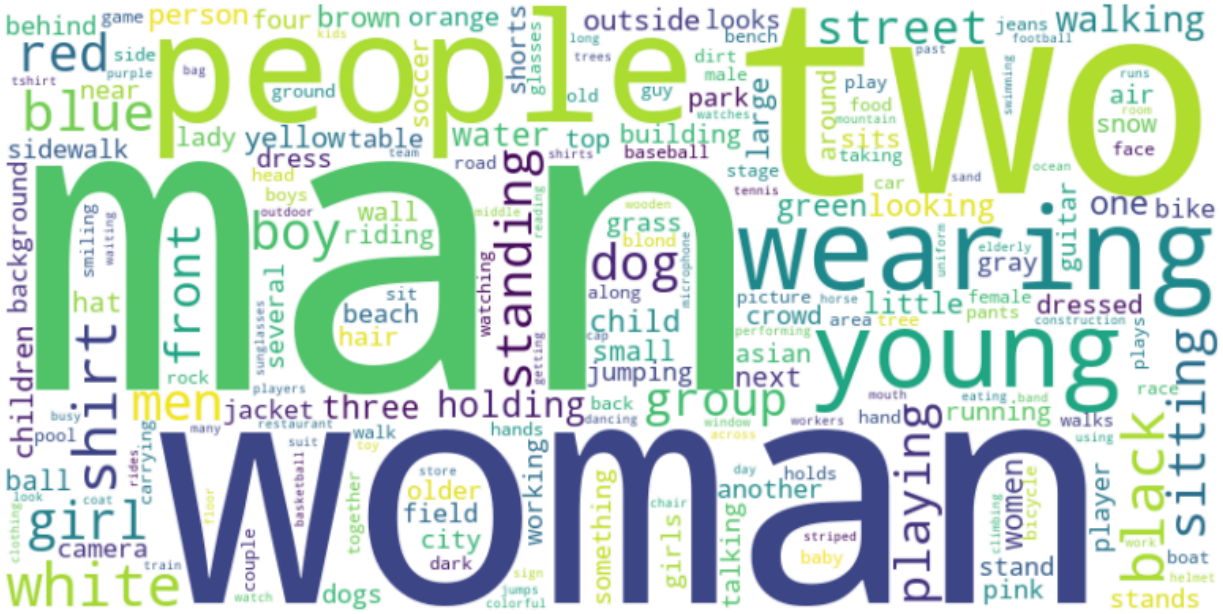

Figure 3

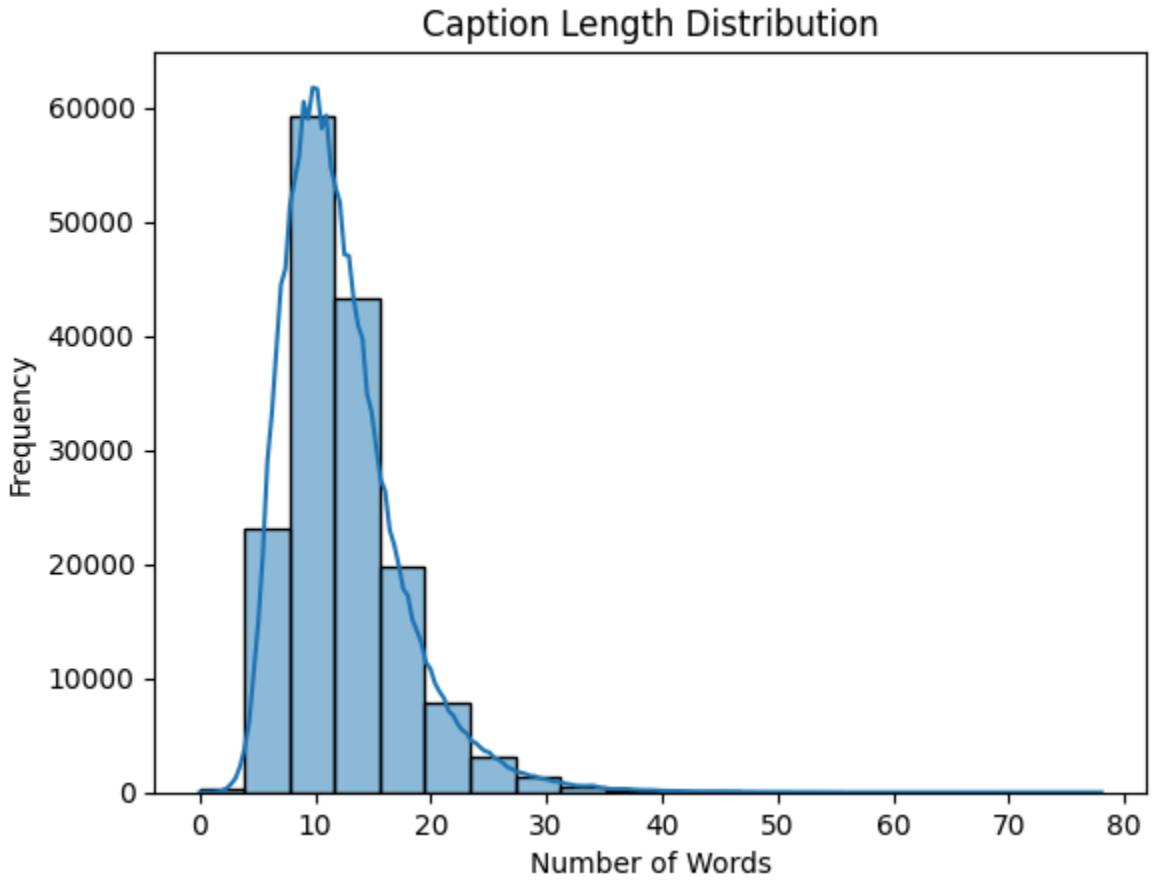

Figure 4

	image	average_similarity
0	1000092795.jpg	0.291171
1	10002456.jpg	0.406812
2	1000268201.jpg	0.556299
3	1000344755.jpg	0.701123
4	1000366164.jpg	0.616992
5	1000523639.jpg	0.298670
6	1000919630.jpg	0.694184
7	10010052.jpg	0.591801
8	1001465944.jpg	0.219137
9	1001545525.jpg	0.395164

Overall Average Caption Similarity for First 10 Images: 0.48

Figure 5

Compute	source	criterion	optimizer	scheduler	rouge	meteor	bleu
A100	CLIP	CrossEntropyLoss	Adagrad	StepLR	0.1996	0.1285	2.17E-157
A100	CLIP	CrossEntropyLoss	Adagrad	CosineAnnealingLR	0.1933	0.1297	1.04E-156
A100	CLIP	CrossEntropyLoss	Adagrad	StepLR	0.1663	0.1287	1.53E-156
A100	CLIP	CrossEntropyLoss	Adagrad	CosineAnnealingLR	0.1604	0.1226	5.53E-157
A100	CLIP	CrossEntropyLoss	Adagrad	ExponentialLR	0.1453	0.0952	9.77E-232
Local	BERT	CrossEntropyLoss	Adagrad	ExponentialLR	0.1615	0.0385	1.94E-233
Local	BERT	CrossEntropyLoss	Adagrad	CosineAnnealingLR	0.1615	0.0385	1.94E-233
Local	BERT	CrossEntropyLoss	RMSprop	ExponentialLR	0.0885	0.0707	6.98E-156
Local	BERT	CrossEntropyLoss	Adagrad	StepLR	0.0871	0.0671	9.21E-156
Local	BERT	CrossEntropyLoss	Adagrad	ExponentialLR	0.0815	0.0659	4.99E-156
Local	CLIP	CrossEntropyLoss	Adagrad	StepLR	0.1943	0.0693	8.9E-232
Local	CLIP	CrossEntropyLoss	Adagrad	ExponentialLR	0.1067	0.0951	9.09E-156
Local	CLIP	CrossEntropyLoss	Adagrad	ExponentialLR	0.1245	0.0504	8.16E-232
Local	CLIP	CrossEntropyLoss	Adagrad	CosineAnnealingLR	0.0884	0.0713	5.97E-156
Local	CLIP	CrossEntropyLoss	Adagrad	StepLR	0.0864	0.0696	5.97E-156
Local	RoBERTa	CrossEntropyLoss	Adagrad	StepLR	0.1574	0.0918	1.61E-155
Local	RoBERTa	CrossEntropyLoss	Adagrad	ExponentialLR	0.1615	0.0516	3.89E-156
Local	RoBERTa	CrossEntropyLoss	Adagrad	ExponentialLR	0.1572	0.0465	7.48E-232
Local	RoBERTa	CrossEntropyLoss	Adagrad	StepLR	0.1161	0.0847	9.26E-232
Local	RoBERTa	CrossEntropyLoss	Adagrad	CosineAnnealingLR	0.1448	0.0433	1.36E-233

Figure 6

image	blip_caption	bleu	rouge1	rouge2	rougeL	meteor
1000092795.jpg	a man standing in the grass	1.2543E-154	0.211363636	0.105714286	0.186363636	0.143995949
10002456.jpg	a metal tower	5.0925E-232	0.171284271	0	0.171284271	0.062389798
1000268201.jpg	a little girl in a pink dress	1	0.498082707	0.314403244	0.498082707	0.352756204
1000344755.jpg	a man on a ladder	0.548811636	0.513899228	0.322709447	0.513899228	0.252345199
1000366164.jpg	two men in a kitchen	0.818730753	0.408724609	0.245990267	0.408724609	0.283218716
1000523639.jpg	a man wearing a black shirt	9.0138E-155	0.194285714	0.055555556	0.194285714	0.081585082
1000919630.jpg	a man sitting on a chair	0.432982015	0.5	0.286967419	0.5	0.336750157
10010052.jpg	a woman on a skateboard	1.1549E-231	0.274661163	0	0.274661163	0.107457717
1001465944.jpg	a group of people standing on a sidewall	7.7115E-155	0.275914787	0.033333333	0.275914787	0.162727776
1001545525.jpg	a man doing a trick on a skateboard	1.2882E-231	0.112822341	0	0.112822341	0.053191312