Exploring Deep Learning Architectures for Image Captioning

Harshit Sampgaon
sampgaon.h@northeastern.edu
Omkar Narkar
narkar.o@northeastern.edu
Sai Dheeraj Malkar
malkar.s@northeastern.edu

Abstract

Image captioning is a critical task in computer vision, aiming to generate descriptive text for images. This project attempts to build understanding for image captioning techniques by comparing the performance of Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models as decoders for a CNN based encoder. Evaluation metrics, including METEOR and BLEU scores, were used to assess the models' performance. Results indicated that the LSTM model achieved a slightly higher METEOR score, while the GRU model scored marginally better in BLEU. This project shows how effective deep learning techniques are for generating image captions and sets the stage for future enhancements by using more advanced encoder models and attention mechanisms.

1 Introduction

1.1 Overview of Project

Image captioning is a critical task in computer vision, aiming to generate descriptive text for images. This project focuses on advancing image captioning techniques using deep learning models. Specifically, we employ the MS COCO 2017 dataset to compare the performance of Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models. The ResNet50 model, pretrained on large image datasets, is utilized for feature extraction, which serves as input for both LSTM and GRU networks to produce captions. By evaluating these models, we aim to determine which approach yields better descriptive accuracy and relevance.

1.2 Motivation

Automatic image captioning has significant implications for accessibility, search, and media management. It plays a vital role in enhancing user experience by enabling machines to understand and describe visual content in natural language. Despite considerable progress, achieving high-quality and contextually accurate captions remains challenging due to the complexity of visual understanding and linguistic generation. Evaluating and comparing LSTM and GRU models helps address these challenges and contributes to the development of an effective captioning systems.

2 Background

Generating natural language descriptions from images has evolved significantly over the years. Initially, most work focused on analyzing videos. Early systems combined visual recognizers with formal language frameworks and rule-based text generation methods. These methods, such as And-Or

Graphs or logic systems, were manually designed and typically applied to specific areas like traffic scenes or sports.

Recently, research has shifted to generating captions for still images. Advances in object recognition—including detecting objects, their attributes, and spatial relationships—have led to new models that create descriptive text for single images. Alam et al. [1] reviewed deep learning methods for image captioning, highlighting the effectiveness of combining Convolutional Neural Networks (CNNs) with Long Short-Term Memory (LSTM) networks. Their study showed that CNNs, such as Inception-v3 and ResNet50, are excellent for extracting features from images, while LSTMs are effective for handling sequential data, resulting in high-quality captions.

The Neural Image Caption (NIC) model introduced by Vinyals et al. [2] represented a major advance in the field. This model uses a CNN for feature extraction and an LSTM for generating captions. It captures complex relationships between objects and their attributes in images. Trained on large datasets like Pascal, Flickr30k, and COCO, the NIC model achieved state-of-the-art results, setting new benchmarks and significantly outperforming previous models in terms of BLEU scores.

Wang et al. [3] expanded deep learning methods to remote sensing image captioning with a word-sentence framework. Their approach includes a word extractor and a sentence generator, specifically designed for the challenges of remote sensing imagery. This model effectively translates high-resolution visual data into coherent textual descriptions, showcasing the versatility of deep learning techniques in various image captioning applications.

Our project builds on using a similar encoder-decoder architecture for image captioning. We utilize a pre-trained ResNet-50 model as the encoder for feature extraction and explore both LSTM and GRU networks as decoders. This approach aims to improve the accuracy and relevance of generated image descriptions by leveraging the strengths of both image analysis and sequential modeling.

3 Approach

3.1 Data Preparation

The first step in our image captioning project is data preparation, which involves loading and pre-processing data from the MS COCO 2017 dataset for image captioning task. We utilized a data loader to handle the batch processing of images and captions. The data loader is initialized with several key parameters, such as `transform`, `mode`, `batch_size`, `vocab_threshold`, and `vocab_from_file`. These parameters control how images are pre-processed, the mode of the data loader (training or testing), the batch size, the minimum word count threshold for vocabulary, and whether to load the vocabulary from an existing file or build vocab from all the training captions.

To pre-process the images, we applied a series of transformations including resizing, random cropping, horizontal flipping, conversion to tensors, and normalization. These transformations ensure the images are in the correct format for the CNN encoder.

Captions were pre-processed by tokenizing each caption, converting tokens to lowercase, and adding special start (<start>) and end (<end>) tokens. These tokens were then mapped to unique numerical indices using a vocabulary built from the training captions. Words appearing fewer than a specified threshold number of times were excluded from the vocabulary to reduce noise. The captions were then converted into sequences of numerical indices, which could be efficiently processed by the RNN decoders.

3.2 Encoder-Decoder Architecture

The core of our image captioning system is based on the encoder-decoder architecture, which is widely used for such tasks. The encoder-decoder framework consists of two main components: the encoder, which processes the input images and transforms them into a context-rich representation, and the decoder, which generates the output captions from this representation.

3.2.1 Encoder: ResNet-50

We utilized a pre-trained ResNet-50 model as the encoder to extract high-level features from the input images. ResNet-50, trained on the ImageNet dataset, is known for its robust feature extraction

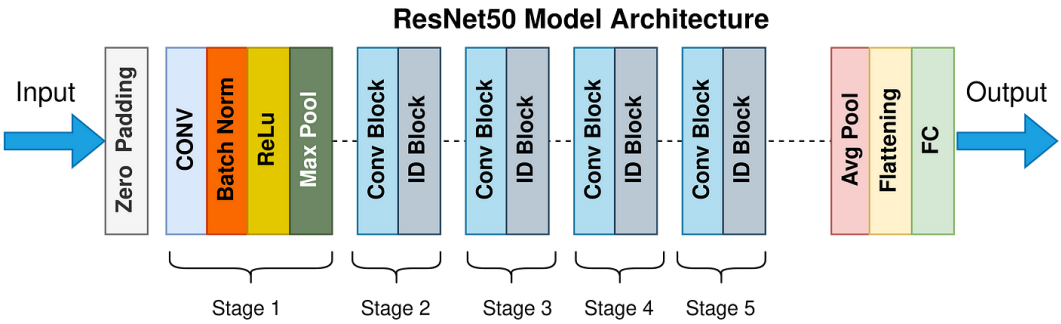

Figure 1: ResNet 50 Encoder Architecture

capabilities due to its deep architecture and residual connections. These residual connections help model the differences before and after the convolution, making ResNet-50 highly effective in tasks like image recognition.

For our purposes, we modified ResNet-50 by removing the final fully connected layer and replacing it with a linear layer. This linear layer transforms the extracted features into a fixed-size embedding vector, which serves as a compact representation of the image. This embedding vector encapsulates the salient features of the image and enables efficient processing by the decoder.

3.2.2 Decoders

Once the image features are extracted and embedded, they are passed to the decoder, which generates captions word by word. We implemented two types of RNN-based decoders to compare the performance of both these types of decoders.

- **LSTM Decoder**: LSTMs are well-suited for sequence modeling tasks due to their ability to capture long-term dependencies. The LSTM decoder architecture includes:
  - An embedding layer that maps each word in the vocabulary to a dense vector representation.
  - An LSTM layer that processes the sequence of embeddings and the initial image features to generate hidden states.
  - A linear layer that transforms the hidden states from the LSTM into vocabulary scores for each time step.
- **GRU Decoder**: GRUs, similar to LSTMs, can capture long-term dependencies but with a simpler architecture and potentially fewer parameters. The GRU decoder consists of:
  - An embedding layer that maps each word in the vocabulary to a dense vector representation.
  - A GRU layer that processes the sequence of embeddings and the initial image features to generate hidden states.
  - A linear layer that transforms the hidden states from the GRU into vocabulary scores for each time step.

3.3 Training and Evaluation

Both models were trained and evaluated using the MS COCO dataset, which contains a large collection of images with associated captions. The dataset provided a robust foundation for training the models to associate image features with corresponding captions, optimizing their ability to generate accurate and contextually relevant descriptions.

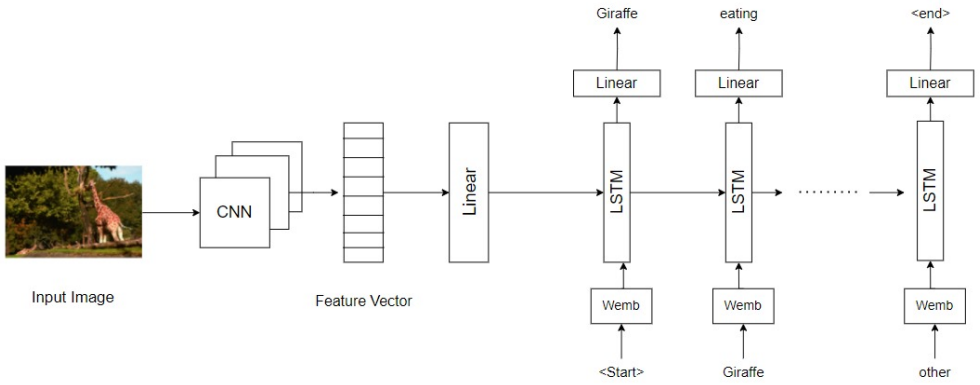

Figure 2: Model Architecture: CNN Encoder + RNN Decoder

4 Results

4.1 Dataset

MS COCO Dataset 2017
To build and evaluate our image captioning model, we utilized the MS COCO dataset 2017. The MS COCO dataset is a large-scale object detection, segmentation, and captioning dataset. For our project, we focused on the image captioning aspect, leveraging the extensive collection of images paired with descriptive captions.

The dataset is divided into several subsets, among which we used the following:

- **train2017**: This subset consists of approximately 118,000 images along with their corresponding captions. This subset was used for training our models.
- **val2017**: This subset consists of approximately 5,000 images and their corresponding captions. This subset was used for validating the performance of our models during training.

The MS COCO dataset provides a diverse set of images and captions, making it an ideal choice for training image captioning models. The extensive variety in the dataset ensures that the models are exposed to a wide range of objects, scenes, and contexts, enhancing their ability to generate accurate and contextually relevant captions.

4.2 Experiments and Performance Evaluation

4.2.1 Experimental Setup

We trained and tested two types of RNN-based decoders: Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU). Both decoders used the same CNN encoder, a pre-trained ResNet-50 model, to extract image features.

4.2.2 Hyperparameters

The following hyperparameters were used in our experiments:

- **Batch Size**: 256
- **Embedding Size**: 256 (dimensionality of image and word embeddings)
- **Hidden Size**: 512 (number of features in hidden state of the RNN decoder)
- **Number of Epochs**: 5
- **Optimizer**: Adam optimizer was used with a learning rate of 0.001
- **Loss Function**: Cross-Entropy Loss was used as the loss function

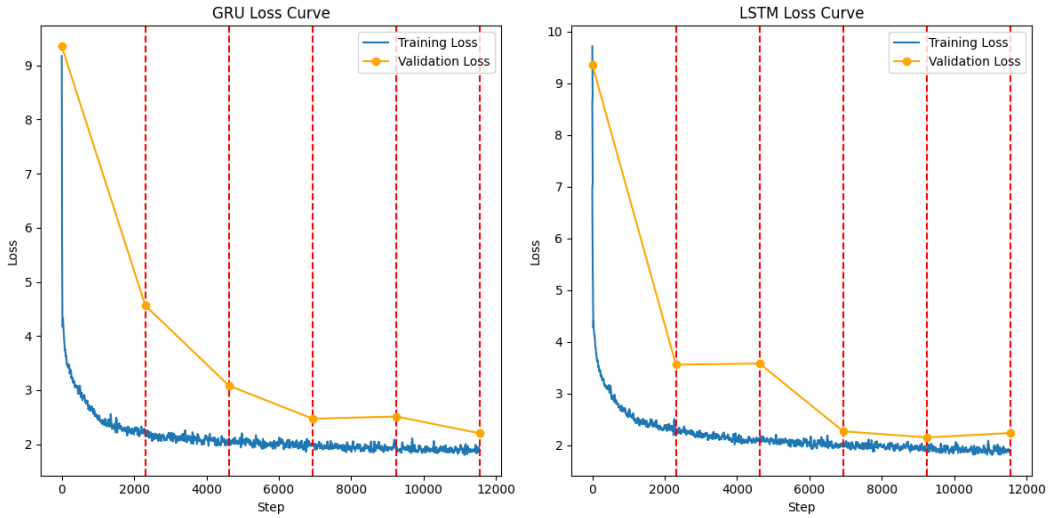

Figure 3: Loss Curve while training for 5 epochs

- **Vocabulary Size**: 11543 (number of unique tokens in the dataset)
- **Sequence Length**: Variable, based on the caption lengths in the dataset

Both models were evaluated on the MS COCO validation set (val2017) to assess their performance in generating accurate and contextually relevant captions. The results are detailed in the next section.

4.3 Result Description

In this study, we used two widely accepted evaluation metrics for image captioning: METEOR and BLEU scores. These metrics provide a quantitative measure of the quality of the generated captions in comparison to reference captions.

METEOR Score: The METEOR (Metric for Evaluation of Translation with Explicit ORdering) score evaluates the quality of machine-generated translations. It considers precision, recall, and fragmentation to provide a more balanced evaluation. METEOR is known for its higher correlation with human judgment compared to other metrics. In the context of image captioning, the METEOR score helps in assessing how well the generated captions match the reference captions in terms of both content and fluency.

BLEU Score: The BLEU (Bilingual Evaluation Understudy) score is another popular metric used in natural language processing for evaluating machine-generated text. It measures the n-gram overlap between the generated and reference captions, providing a precision-based evaluation of the generated text. BLEU scores are particularly useful for assessing the exactness and adequacy of the generated captions, though they may sometimes miss finer nuances of language quality.

Using these metrics, we evaluated the performance of our image captioning models, as shown in Table 1.

Model	METEOR Score	BLEU Score
GRU	0.2240	**0.2868**
LSTM	**0.2360**	0.2830

Table 1: Performance Comparison of GRU and LSTM Models

These metrics collectively provide a comprehensive evaluation of the models' performance in generating accurate and contextually relevant image captions.

Some **good** results of generated captions:

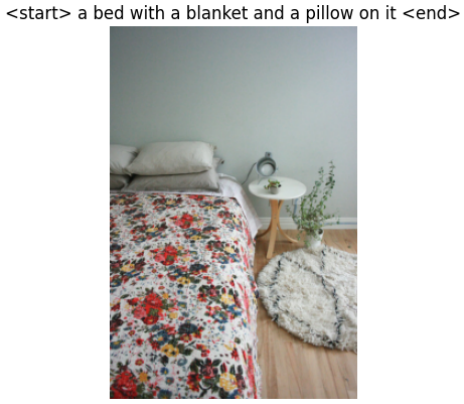

Figure 4: Good Result of objects

Figure 5: Good Result for objects and human

Figure 6: Good Result of objects

Figure 7: Good Result for objects and human

Some **bad** results of generated captions:

Figure 8: Bad Result of objects

Figure 9: Bad Result for objects and human

4.4 Discussions

As observed from the results in Table 1, the LSTM model achieved a slightly higher METEOR score compared to the GRU model, indicating better alignment with human judgment in terms of content and fluency. However, the GRU model achieved a slightly higher BLEU score, reflecting better precision in n-gram overlap with reference captions. To generate captions with better accuracy, we should use an LSTM-based decoder model.

In a broader context, our results suggest that RNN-based decoders, specifically LSTM models, are more effective in generating human-like captions for images. This is consistent with prior research

which highlights the superior capability of LSTM networks in capturing long-term dependencies in sequential data. The slight advantage of the GRU model in BLEU score indicates that it still performs competitively, particularly in scenarios requiring shorter computational times and fewer parameters.

4.5 Future Directions

Future work could explore the use of more advanced encoder CNNs and incorporate attention mechanisms to further enhance model performance. As demonstrated in the "Show, Attend and Tell" paper, attention mechanisms can significantly improve the quality of generated captions by focusing on relevant parts of the image. Implementing such techniques could lead to even better results.

5 Conclusion

In this project, we explored the capabilities of deep learning architectures for the task of image captioning, focusing on comparing Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models. Utilizing the MS COCO 2017 dataset, we leveraged a pre-trained ResNet-50 model for feature extraction, followed by RNN-based decoders to generate captions. Our aim was to assess which model could provide more accurate and contextually relevant descriptions.

Our approach centered on an encoder-decoder architecture, where the ResNet-50 served as a powerful encoder due to its extensive training on the ImageNet dataset. The decoders, LSTM and GRU, were chosen for their proven capabilities in handling sequential data and capturing long-term dependencies.

Through rigorous training and evaluation, the LSTM model demonstrated a slight edge over the GRU model in terms of the METEOR score, indicating better alignment with human judgment. Conversely, the GRU model showed a marginally higher BLEU score, suggesting superior precision in 4-gram overlap.

The key takeaway from our project is that LSTM-based decoders tend to generate more human-like captions, confirming their effectiveness in sequence modeling tasks. However, the competitive performance of the GRU model highlights its potential, especially in scenarios demanding faster computation and fewer parameters.

In conclusion, our work explored research in image captioning, reinforcing the value of deep learning techniques in enhancing the accuracy and relevance of generated captions. Future work could build upon these models by integrating more advanced encoder architectures and attention mechanisms.

References

[1] Alam, et al. "Deep learning to undertake an analysis of image captioning." 2022 International Conference on Computer Communication and Informatics (ICCCI), Jan. 25–27, 2022, Coimbatore, India. DOI: 10.1109/ICCCI54379.2022.9740788.

[2] Vinyals, et al. Show and Tell: A Neural Image Caption Generator. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 3156-3164. DOI: 10.1109/CVPR.2015.7298935

[3] Wang, et al. "Remote Sensing Image Captioning through Word-Sentence Framework." IEEE Transactions on Geoscience and Remote Sensing, vol. 60, no. 4, April 2022. DOI: 10.1109/TGRS.2022.3145983.