Project Report: Image Captioning Using CNN+LSTM and Transformers

By: Shivangi Modi (20093), Shruti Kavishwar (20022)

Guided By: Prof. Andrei Grigoriev

San Francisco Bay University

Abstract

This project explores image captioning using two distinct approaches: a CNN+LSTM-based sequence model inspired by the *'Show and Tell: A neural Image Caption Generator'* paper and a transformer-based architecture inspired by the *'Attention is All You Need'* paper. Using the Flickr8k dataset, the models were implemented and compared in terms of their quantitative performance (BLEU scores) and qualitative outputs (example captions). The transformer model demonstrated a BLEU score of 0.75 and produced more accurate and descriptive captions, while the CNN+LSTM model achieved a BLEU score of 0.65 and generated captions that were less specific.

1. Introduction

Image captioning bridges the fields of computer vision and natural language processing, enabling machines to generate textual descriptions of images. The primary motivation for this project is to compare the performance of traditional sequence-based methods (CNN+LSTM) with attention-based transformer architectures in the context of generating captions for images. The Flickr8k dataset was chosen for its moderate size, enabling computational feasibility while providing sufficient variety for evaluation.

2. Related Work

2.1. 'Show and Tell: A Neural Image Caption Generator'

This foundational work proposed a sequence-to-sequence approach for image captioning. It uses a convolutional neural network (CNN) to extract image features, which are then fed into a recurrent neural network (RNN) to generate captions sequentially. While effective, it lacks explicit mechanisms to focus on specific parts of the image when generating captions, making it less robust for complex datasets.

2.2. 'Attention is All You Need'

This paper introduced the transformer architecture, revolutionizing sequence modeling by replacing recurrence with self-attention mechanisms. Its ability to model long-range dependencies and handle sequence data efficiently has been adapted to various tasks, including image captioning.

The incorporation of self-attention in transformers provides a more global view of the input, significantly improving caption generation quality.

3. Proposed Method

3.1. Formulation

The image captioning task is formulated as a sequence generation problem where the input is an image, and the output is a sequence of words forming a caption.

- **CNN+LSTM Approach**: Features are extracted using MobileNetV3Small, followed by a fully connected layer and passed into an LSTM to generate captions.
- **Transformer Approach**: A transformer encoder-decoder architecture is used, inspired by the *'Attention is All You Need'* paper.

3.2. Implementation

3.2.1. Transformer-Based Implementation

The transformer-based implementation is inspired by the *'Attention is All You Need'* paper. The model includes an encoder for feature extraction and a decoder for generating captions. The encoder uses multi-head self-attention mechanisms to process image features, while the decoder generates captions based on the image embeddings and previously generated tokens.

3.2.2. CNN+LSTM Implementation

The CNN backbone (MobileNetV3Small) extracts image features, which are pooled and passed through a dense layer for dimensionality reduction. The features are then sequentially processed by an LSTM to generate captions. The LSTM outputs predictions based on the tokenized captions.

3.3. Data Preprocessing

3.3.1. Image Preprocessing

Images were resized to a fixed size (224, 224, 3), normalized by dividing pixel values by 255, and augmented using random horizontal flips, brightness adjustments, and rotations to improve model generalization.

3.3.2. Caption Preprocessing

Captions were tokenized using a TextVectorization layer, converted to lowercase, and stripped of punctuation. Special tokens [START] and [END] were added to each caption for sequence demarcation.

3.4. Model Architectures

The two architectures differ significantly in their approach to modeling sequence generation and handling attention:

- **CNN+LSTM Architecture**:
  - Sequentially processes input features from the CNN using LSTM layers.
  - Lacks an explicit attention mechanism, limiting its ability to selectively focus on specific image regions during caption generation.
  - Computationally less intensive but constrained by the sequential nature of RNNs.
- **Transformer Architecture**:
  - Employs multi-head self-attention to model long-range dependencies.
  - The encoder-decoder structure allows the model to focus on relevant image regions while decoding captions.
  - Highly parallelizable, resulting in faster training times but with higher computational requirements.

3.5. Experiments

3.5.1. Evaluation

The performance of both models was evaluated using BLEU scores as the primary quantitative metric. Additionally, qualitative analysis was conducted by comparing example captions generated for test images.

3.5.2. Comparison

3.5.2.1. Quantitative Results

- **CNN+LSTM Model**: BLEU Score = 0.65

```
# Example: Evaluating BLEU on the test set

average_bleu = evaluate_bleu(encoder, decoder, vectorizer, test_images, test_captions)
print(f"Average BLEU Score: {average_bleu}")
```

```
Average BLEU Score on Test Set: 0.6552575111476422
Average BLEU Score: 0.6552575111476422
```

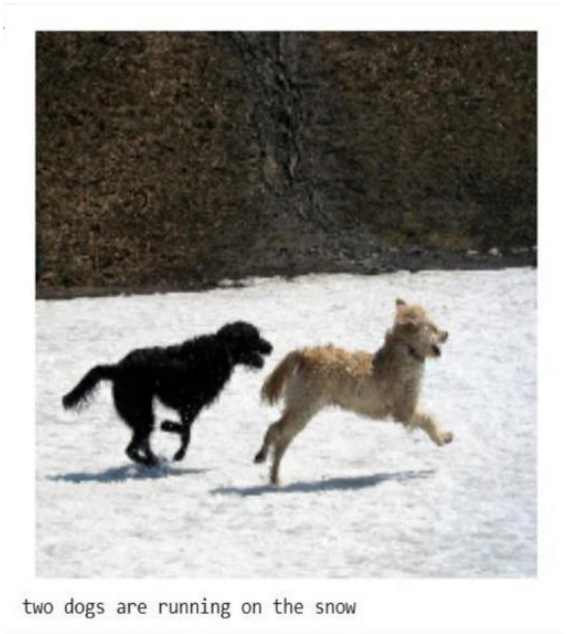
- **Transformer Model**: 0.75.

3.5.2.2. Qualitative Analysis

- **CNN+LSTM Model**:
  - Captions were generally coherent but lacked specificity for complex images.

- Example Output: A black dog is jumping in

- **Transformer Model**:
  - Captions were more descriptive and contextually accurate.
  - Example Output: Two dogs are running on the snow.



3.6. Conclusion

The project demonstrated the effectiveness of transformer-based models in generating high-quality captions for images, outperforming the traditional CNN+LSTM architecture in both quantitative metrics and qualitative analysis. The main challenges encountered included computational limitations, which restricted the ability to experiment with larger datasets or more complex transformer variants.

Future Work

Future enhancements could include:

- Exploring larger and more diverse datasets such as COCO or Flickr30k.
- Implementing attention mechanisms in the LSTM model to improve its performance.
- Optimizing the transformer architecture for better efficiency on limited hardware.