Image Captioning Using Flickr 8k Dataset

Anish Shivram Gawde, Harshil Mistry
gawde.ani@northeastern.edu, mistry.har@northeastern.edu
Northeastern University, Khoury College of Computer Sciences, Boston, MA
DS5220: Supervised Machine Learning and Learning Theory
Professor Hongyang Zhang
December 9, 2024

Abstract

Image Captioning is a multidisciplinary task in artificial intelligence that combines computer vision and natural language processing to generate textual description of visual content. It has a wide range of applications, including making visual media accessible to the visually impaired, improving search engine systems and countless more. With advancements in deep learning, modern approaches use approaches like an encoder-decoder system that combine Convolutional Neural Networks for visual feature extraction and LSTM or transformers for generating text. Emerging models like Vision Transformers (ViT), GPT-based decoders, and CLIP (Contrastive Language-Image Pretraining) have revolutionised this task by enabling deeper understanding and are generating captions which align much more closely with the images. These systems aim to understand the content and context of images while producing grammatically and semantically accurate captions. Despite significant progress, challenges remain, including managing dataset biases, generating diverse and meaningful descriptions, and balancing accuracy with creativity. This report dives into the techniques, tools, and challenges of building effective image captioning models, emphasizing the seamless integration of visual and linguistic elements.

Overview

This project aims to develop a system capable of producing image captions similar to those created by humans. This involves training the model to understand the content of an image, including object detection, understanding the actions and being aware of the context. It also involves using these understandings to generate meaningful, grammatically and semantically correct captions. These captions also need to be descriptive and should accurately convey the meaning of the image.

This problem is particularly interesting because it addresses the broader challenge of enabling AI to understand and describe the visual world. The motivation behind this project is bridging computer vision and natural language processing; two very important aspect of artificial intelligence and also its potential to address real world problems such as:

- **Medical Image Analysis**: Describing medical images like X-rays or CT scans to aid diagnosis.
- **Food Recognition and Calorie Estimation**: Analyzing food images to provide nutritional information.
- **Accessibility Enhancements**: Making visual content accessible to visually impaired individuals

This problem is not just a technical challenge but also a creative one. It involves understanding nuances, relationships, and the interplay between objects, actions, and context. This is what makes the problem both intriguing and impactful, pushing the boundaries of AI to create systems that better understand and interact with the world around us.

Proposed Approach

To create an effective image caption generator, this project uses state-of-the-art deep learning technologies for both image feature extraction and natural language generation. Our approach focuses on two main components: [6] **Image Feature Extraction**, and **Caption Generation.**

- **Image Feature Extraction:** This project uses [3] **ResNet50**, a pre-trained Convolutional Neural Network (CNN), to extract high-level features from images. ResNet50, trained on the ImageNet dataset offers robust and meaningful visual representations without requiring extensive retraining. The pre-trained model eliminates the need for large image datasets specific to captioning, significantly reducing resource requirements [6].

- **Caption Generation:** Implementing an [5] LSTM-based model to generate captions from the extracted features. LSTMs are well-suited for sequence generation tasks due to their ability to handle temporal dependencies and retain information over long sequences. This mechanism in LSTMs allows the model to focus on relevant parts of the input features while maintaining context throughout the sentence generation process. They are highly adaptable to various datasets and tasks, making them a reliable choice for caption generation in diverse scenarios.

The combined architecture integrates ResNet50 for feature extraction and an LSTM-based model for decoding, providing a scalable and efficient solution to image captioning.

Limitations:

- **Dataset Size**: Smaller datasets like Flickr8k may limit the generalization of the model to diverse or complex image scenarios.
- **Bias**: Captions might reflect dataset biases, leading to repetitive patterns or stereotypical descriptions.
- **Complex Contexts**: The system may struggle with abstract or nuanced relationships in highly complex images.

The proposed approach leverages [5] LSTMs, which handle long-term dependencies more effectively and are computationally efficient for sequential data tasks. Models like CLIP and ViT require large-scale pretraining on massive datasets, making them resource-intensive. The proposed approach using ResNet50 is used for image feature extraction due to its efficiency, reliability, and pretraining on ImageNet, which provides robust embeddings for visual content. LSTMs for decoding and ResNet50 for feature extraction strike a balance between performance and resource requirements.

Experiment Setup

The dataset utilized for this project is the widely used [1] **Flickr8k dataset**, a benchmark dataset in the field of image captioning. It is specifically designed to bridge visual content and descriptive language, providing a valuable resource for training and evaluating models that generate captions for images.

The dataset consists of **8,091 images**, each carefully curated to include a variety of visual scenarios, from indoor settings like living rooms and kitchens to outdoor scenes such as parks, beaches, and cityscapes. The images capture a diverse range of objects, activities, and interactions, making it well-suited for building generalized models that can understand and describe real-world images.

Each image is paired with **five unique captions**, resulting in a total of **40,455 captions**. These captions were written by human annotators and provide a rich diversity in how visual content can be described. This variation helps models learn to generate captions that are not only accurate but also contextually nuanced. The captions range in length, typically averaging **10–15 words**, and are presented in plain text format. They focus on describing key elements in the image, including objects, actions, and the surrounding context.

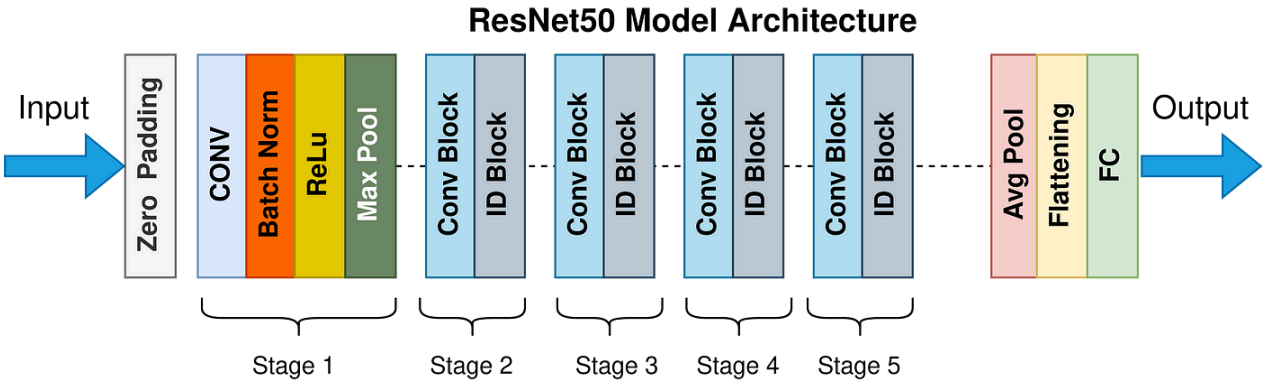
Sample Images from the Dataset

Implementation

Models:

- **Feature Extraction**

  For feature extraction, we use **ResNet50**, a pre-trained convolutional neural network (CNN) known for its robust feature representation capabilities. Trained on the ImageNet dataset, ResNet50 effectively captures high-level semantic features from images, such as objects, textures, and contextual relationships.



Source: Towards Data Science

Input Processing: The image is padded, processed through a convolution layer to extract basic features, normalized using batch normalization, and activated with ReLU for non-linearity.

Pooling: A max pooling layer reduces the image size while keeping important information.

Residual Blocks: The main part consists of:

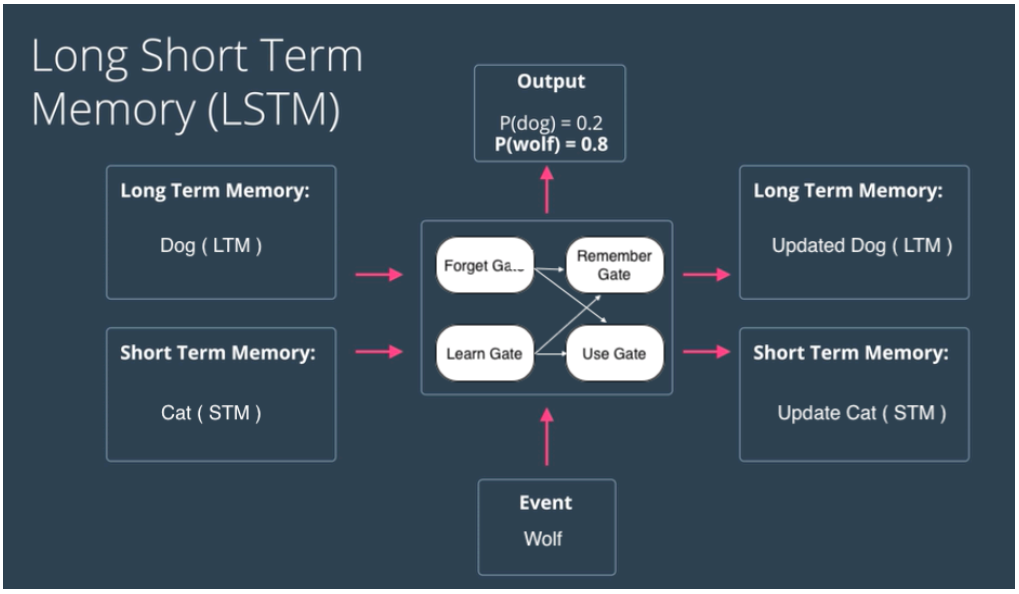
- **Conv Blocks**: Adjust feature sizes with shortcut connections.
- **Identity Blocks**: Retain input size and help the network focus on learning new patterns while keeping previous information.

Feature Aggregation: An average pooling layer condenses the features, which are then flattened into a single vector.

Modification: The architecture is modified to output features from the **penultimate layer** instead of the final classification layer. This approach ensures that the extracted features can be adapted for tasks requiring a higher level of abstraction, like generating captions or performing object detection, rather than being restricted to classifying images into predefined categories.

- **Caption Generation**

For caption generation, we use a **Long Short-Term Memory (LSTM)** network enhanced with an attention mechanism. The LSTM is well-suited for handling sequential data, and attention adds the ability to focus on specific parts of the image while generating captions, improving both context understanding and output quality.



Source: Analytics Vidya

An **LSTM (Long Short-Term Memory)** network is a type of neural network that's really good at understanding sequences and remembering important information over time. LSTM's have kind selective memory, they hold on to what is important and forget what they deem unimportant. It has two types of memory: **long-term memory (LTM)**, which stores things you want to remember for a long time, and **short-term memory (STM)**, which handles temporary information. At each step, the LSTM uses **gates** to decide what to keep, what to forget, and what new information to learn. For example, the **Forget Gate** drops irrelevant information from the LTM, while the **Learn Gate** adds useful new information. The **Use Gate** focuses on combining STM and LTM outputs to produce the final output. This process ensures that the LSTM captures the right context for each step, making it ideal for tasks like language generation or time-series analysis where understanding the flow of information is key.

Architecture and Workflow

1. Image Feature Processing

- **Input**: Pre-extracted features of shape (2048,) from ResNet50 represent high-level image semantics.
- **Normalization**: A BatchNormalization layer which ensures feature scaling and making training more efficient.
- **Dense Transformation**: The features are passed through a Dense layer with 512 neurons and a ReLU activation function, reducing dimensionality while retaining important information. It also has a L2 regularization to help reduce overfitting.

2. Text Sequence Processing

- **Input**: Tokenized captions, padded to a fixed max_length, are provided as input.
- **Embedding**: An embedding layer converts each word index into a dense 256-dimensional vector, capturing semantic meaning.
- **Normalization**: A BatchNormalization layer normalizes the embeddings to improve model stability.
- **Bidirectional LSTM**: A bidirectional LSTM processes the embeddings, capturing dependencies in both forward and backward directions. This helps the model understand word context more effectively.

3. Attention Mechanism

- **Attention Scores**: Computes the similarity between repeated image features and LSTM outputs using a Dot layer.
- **Attention Weights**: A softmax activation normalizes the scores into probabilities, which emphasize the most relevant parts of the image.
- **Context Vector**: A weighted sum of LSTM outputs, capturing image details crucial for generating the current word.

4. Decoder

- **Flattening**: This layer converts the context vector into a single-dimensional array.
- **Feature Combination**: This layer concatenates the flattened context vector with image features, integrating visual and textual data.
- **Decoding**: This processes the combined features using a Dense layer with 512 neurons and ReLU activation.
- **Output Layer**: A Dense layer with softmax activation predicts the probabilities for each word in the vocabulary.

Compilation

- **Loss Function**: categorical_crossentropy compares predicted word probabilities to true values.
- **Optimizer**: Adam optimizer (learning rate: 5e-4)
- **Metrics**: Accuracy is used to evaluate model performance.

Computing Environment:

a) **GPU**: NVIDIA Tesla T4

b) **Frameworks**: TensorFlow/Keras for model implementation and training

c) **Platform**: Google Colab or local system with CUDA-enabled GPU support

Experiment Results

In this section, we will discuss how our model performed in generating captions for images. The evaluation includes quantitative metrics like BLEU scores, which measure how closely the generated captions match the reference captions, and qualitative examples that showcase the model's ability to describe image content accurately.

Evaluation Metrics

To assess the model's performance quantitatively, we use [4] BLEU (Bilingual Evaluation Understudy) scores. BLEU scores evaluate the similarity between the generated captions and reference captions by comparing n-grams (sequences of words) in both. Higher BLEU scores indicate better alignment with the reference captions, with BLEU-1 measuring single-word matches, BLEU-2 for two-word pairs, and so on up to BLEU-4.

Example Image 1

Generated Caption

```
young girl with pigtails painting outside in the grass
```

```
BLEU score for image 3.jpg: 0.7364
```

Example Image 2

Generated Caption

```
a young girl in a white top is looking down inside a tunnel smiling
```

```
BLEU score for image 7900.jpg: 0.6773
```

The model is able to generate captions that are semantically relevant, grammatically correct, and aligned with the visual content of the images. It successfully identifies objects, actions, and contexts in the images and translates them into descriptive sentences. For example it is able to identify a little girl painting outside and matches the reference captions closely. However, there are instances where the model generates overly generic captions or misses subtle relationships between objects, indicating room for improvement in handling complex or nuanced scenes.

Example Image 3

```
a woman wearing a red and white striped shirt and a little boy standing on the side of a road
```

```
BLEU score for image 451.jpg: 0.4287
```

The model generates a grammatically correct caption but it fails to understand the objects in the image properly. This indicates significant room for improvement, particularly in the model's ability to comprehend and differentiate objects accurately. Enhancements could involve training on a larger, more diverse dataset, fine-tuning the feature extractor, or incorporating a more advanced vision-language model like CLIP or ViT. Since the [1] Flickr8k dataset has only 8091 images, the model is limited by the dataset's size and diversity, which impacts its ability to generalize to a wide variety of objects and scenes. Datasets like MSCOCO and Flickr30k could be used for further training and improving the model.

Model analysis on custom images

Analysing the models performance on custom images is a tricky task, as the custom images have no captions associated with them. As a result, the model has no reference captions to help generate a new caption. Metrics like [4] BLEU Scores can also not be used since there are no ground-truth captions. Nevertheless, the proposed approach works on custom images as well, which is:

1. ResNet50 is used to extract the image features.
2. These features are given to the [5] LSTM decoder as input which then generates the captions.

While visual inspection of captions is the primary method for assessing performance on custom images, the results indicate that the model can describe general features and contexts but may struggle with object-specific nuances due to dataset limitations.

Example Image 1

```
man wearing a brown jacket is sitting on a rock along a winding river with trees behind
```

This is an example where the model successfully understood the objects and their relationships when given a custom image. However, in most cases, the model struggled to do so, highlighting its limitations. As discussed earlier, training the model on a more diverse dataset and fine-tuning it is bound to improve the caption generation.

Example Image 2

```
Generated Caption: a black and white dog in a dark room barking in the sun
```

In this particular example, the model misidentifies a cat as a dog, likely due to the dataset's bias, as it contains a larger number of images featuring dogs. This highlights the model's reliance on dataset distribution and underscores the need for a more balanced and diverse dataset to improve object recognition accuracy.

Future Improvements

While the current model demonstrates the ability to generate captions, there is significant room for enhancement. Key areas for future improvements include:

- Train the model on larger and more diverse datasets, such as MS COCO, to improve its ability to generalize across various objects and scenes.
- Improve the attention mechanism to dynamically focus on more relevant parts of the image for each generated word, ensuring better alignment between visual and textual features.
- Add regularization methods like dropout, L2 regularization, or early stopping to improve model generalization and reduce overfitting.
- Use advanced data augmentation techniques to increase the variety of training samples and reduce overfitting.

Comparison with existing solutions

Compared to existing solutions, our model offers a balanced approach between efficiency, simplicity, and effectiveness for image captioning. Many state-of-the-art models rely on advanced architectures like CLIP or ViT, which require extensive pretraining on massive datasets and significant computational resources. In contrast, our model leverages [3] ResNet50, a pre-trained and efficient feature extractor, combined with an [5] LSTM-based decoder, which is computationally lighter and more accessible.

Existing models often achieve high performance using evaluation metrics like [4] BLEU However, these solutions may lack practicality due to their complexity and reliance on large-scale data. Our model focuses on using pre-trained ResNet50, which reduces the need for a large dataset while still generating meaningful and contextually accurate captions. The LSTM decoder handles sequential dependencies effectively, ensuring the generated captions are coherent and relevant.

While models using SPICE and CIDEr provide deep semantic understanding, they can be resource-intensive. Our approach strikes a balance by achieving competitive results with fewer computational requirements, making it a scalable and efficient choice for diverse applications. This makes our model a practical alternative for tasks requiring robust caption generation without the need for massive computational resources or datasets.

Conclusion

In this project, we successfully developed an image captioning system that combines the strengths of ResNet50 for image feature extraction and an LSTM-based model for generating descriptive captions. The system demonstrated its ability to produce coherent and contextually relevant captions for images in the Flickr8k dataset. By leveraging pre-trained ResNet50, we reduced the need for extensive data and computational resources while still achieving robust visual feature representation. The LSTM-based architecture effectively handled sequential dependencies, allowing for natural and meaningful language generation. This work highlights the potential of combining pre-trained visual models with sequence-based language models to tackle challenging tasks like image captioning, with scalable applications in real-world scenarios.

References

[1] **Flickr8k Dataset**:Hodosh, M., Young, P., & Hockenmaier, J. (2013). Framing Image Description as a Ranking Task: Data, Models and Evaluation Metrics. *Journal of Artificial Intelligence Research*, 47,853-899.
https://www.kaggle.com/code/phyosandarwin/efficientnet-lstm-model

[2] **Image captioning:** S. Yan, Y. Xie, F. Wu, J. S. Smith, W. Lu and B. Zhang, "Image captioning via hierarchical attention mechanism and policy gradient optimization", *Signal Process.*, vol. 167, Feb. 2020.
https://www.sciencedirect.com/science/article/abs/pii/S0165168419303822?via%3Dihub

[3] **ResNet50**: He, K., Zhang, X., Ren, S., & Sun, J. (2016). "Deep Residual Learning for Image Recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770-778
https://ieeexplore.ieee.org/document/7780459

[4] **BLEU Metric**: Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). "BLEU: A Method for Automatic Evaluation of Machine Translation." Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL), 311-318.
https://aclanthology.org/P02-1040/

[5] **LSTM**: Hochreiter, S., & Schmidhuber, J. (1997). "Long Short-Term Memory." Neural Computation, 9(8), 1735-1780.
https://direct.mit.edu/neco/article-abstract/9/8/1735/6109/Long-Short-Term-Memory?redirectedFrom=fulltext

[6] **Image Feature Extraction:** He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778.
https://ieeexplore.ieee.org/document/7780459