Indian Institute of Information Technology, Nagpur

Image Captioning

Dr. Nishat Ansari, Rohan Udhwani (BT21CSE092), Ansh Bansal (BT21CSE143), Gaurav Bajpai (BT21CSE150), Robin Bansal (BT21CSE077)

Department of Computer Science & Engineering

Abstract— Image captioning is the process of generating descriptions about what is going on in the image. By the help of Image Captioning descriptions are built which explain about the images. Image Captioning is basically very much useful in many applications like analyzing large amounts of unlabeled images and finding hidden patterns for Machine Learning Applications for guiding Self driving cars and for building software that guides blind people. This Image Captioning can be done by using Deep Learning Models. With the advancement of deep learning and Natural Language Processing now it has become easy to generate captions for the given images. In this paper we will be using Neural Networks for the image captioning. Convolution Neural Network (DenseNet) is used as encoder which access the image features and Recurrent Neural Network (Long Short Term Memory) is used as decoder which generates the captions for the images with the help of image features and vocabulary that is built.

Keywords— Deep Learning; Image Captioning; Convolutional Neural Networks; Recurrent Neural Networks; DenseNet; Long Short Term Memory;

Table of Contents

Introduction

What is Image Captioning ?

In the realm of artificial intelligence and computer vision, Image Captioning stands as a captivating intersection where the worlds of Natural Language Processing

(NLP) and Computer Vision converge. At its essence, Image Captioning encapsulates the profound ability of machines to understand and describe the content of images in human-like language. This transformative task holds immense significance, bridging the gap between visual perception and linguistic expression, and unlocking a myriad of applications across diverse domains.

Image Captioning, in its simplest definition, is the process of generating textual descriptions for images. It embodies a symbiotic relationship between Computer Vision techniques, which extract visual features from images, and Natural Language Processing methodologies, which decode these features into coherent and descriptive text sequences. This interdisciplinary endeavor represents a harmonious fusion of cutting-edge technologies, where sophisticated algorithms discern the intricate details of visual content and articulate them into intelligible narratives.

At the heart of most Image Captioning systems lies an encoder-decoder framework, a paradigm that epitomizes the synergy between Computer Vision and Natural Language Processing. Within this framework, an input image undergoes encoding, wherein it is transformed into an intermediate representation encapsulating the salient features of the visual content. Subsequently, this encoded representation serves as the foundation for the decoding phase, where it is translated into a descriptive text sequence that encapsulates the essence of the image.

Throughout this report, we embark on a comprehensive exploration of Image Captioning, traversing the landscape of methodologies, datasets, experimental setups, and performance evaluations. Our journey unfolds amidst a backdrop of innovation and discovery, where we delve into the intricacies of model architectures, training strategies, and empirical results. By dissecting the core principles and practical implementations of Image Captioning, we endeavor to unravel its mysteries and illuminate the path towards further advancements in this captivating field.

Join us as we navigate the realms of Computer Vision and Natural Language Processing, unraveling the complexities of Image Captioning and embarking on a journey of discovery and innovation. Through this endeavor, we aim to shed light on the transformative potential of Image Captioning and pave the way for new frontiers in artificial intelligence and human-machine interaction.

Literature Review

In the method proposed by Liu, Shuang et al. [1], two models of deep learning are discussed: Convolutional Neural Network-Recurrent Neural Network (CNN-RNN) Based Image Captioning and Convolutional Neural Network-Convolutional Neural Network (CNN-CNN) Based Image Captioning. In the CNN-RNN framework, Convolutional Neural Networks (CNNs) are used for encoding, and Recurrent Neural Networks (RNNs) are used for the decoding process. CNNs convert images into vectors known as image features, which are then passed into Recurrent Neural Networks as input. The NLTK libraries are utilized to generate the actual captions for the project. In the CNN-CNN based framework, only CNN is employed for both encoding and decoding of the images. A vocabulary dictionary is utilized, mapped with image features to generate the exact word for the given image using the NLTK library, thus producing error-free captions. Employing multiple models simultaneously in the CNN-CNN model is quicker compared to the sequential training process of the CNN-RNN model. The CNN-CNN model has a shorter training time but may have higher loss compared to the CNN-RNN model.

In the method proposed by Ansari, Hani et al. [2], an encoding-decoding model is used for image captioning, along with two additional models: Retrieval-based captioning and Template-based captioning. Retrieval-based captioning involves placing training images and their corresponding generated captions in separate spaces, calculating correlations for test images and captions, and retrieving the caption with the highest correlation value. Prototype-based description is another technique utilized, where the Inception V3 model serves as the encoder, and attention mechanism and Gated Recurrent Units (GRUs) function as the decoder to generate captions.

In the method proposed by Subrata Das, Lalit Jain et al. [3], the focus is on military image captioning using a CNN-RNN based framework. The Inception model is employed for encoding the images, and Long Short-Term Memory (LSTM) networks are utilized to address gradient descent problems.

In the method proposed by G Geetha et al[4] they have used CNN-LSTM model for image captioning. The entire flow of the model was explained from data set collection to caption generation. Here Convolutional Neural Networks was used as encoder and LSTM’s was used as decoder for generating the captions.

Methodology

1.1 Model Architecture

The CNN-RNN architecture is a powerful framework employed for image captioning tasks, seamlessly blending

Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) into a cohesive pipeline. Within this architecture, CNNs are responsible for feature extraction from images, while RNNs, particularly Long Short-Term Memory (LSTM) networks, handle text generation. This integrated approach leverages both visual and textual information to produce descriptive captions that accurately depict the content of the images.

In the context of feature extraction, CNNs play a pivotal role by analyzing the visual content of images and extracting meaningful features. These networks are composed of various layers, including convolutional and pooling layers, designed to capture hierarchical representations of visual patterns. Pre-trained CNN models such as DenseNet are often utilized to extract high-level features from images, providing rich contextual information for caption generation.

Conversely, LSTMs excel in modeling sequential data and are adept at generating coherent text. Within the CNN-RNN architecture, LSTMs receive the visual features extracted by the CNN as input and iteratively generate words to form captions. By maintaining an internal state and capturing long-range dependencies in the input sequence, LSTMs effectively encode the semantic context of the image and generate descriptive captions that accurately describe its content.

This hybrid architecture seamlessly integrates the strengths of CNNs and LSTMs, allowing for the seamless fusion of visual and textual information. Through joint training on image-caption pairs, the model learns to associate visual features with corresponding textual descriptions, enabling it to generate accurate and contextually relevant captions for new images during inference. The effectiveness of this approach underscores its importance in bridging the gap between computer vision and natural language processing, making it a cornerstone in the field of image captioning.

To perform Image Captioning we will require two deep learning models combined into one for the training purpose
CNNs extract the features from the image of some vector size aka the vector embeddings. The size of these embeddings depend on the type of pretrained network being used for the feature extraction
LSTMs are used for the text generation process. The image embeddings are concatenated with the word embeddings and passed to the LSTM to generate the next word
For a more illustrative explanation of this architecture check the Modelling section for a picture representation


"man in black shirt is playing guitar."


"construction worker in orange safety vest is working on road."


"two young girls are playing with lego toy."

1.2 Data Preprocessing

Steps for preprocessing caption text:

1. Convert sentences into lowercase
2. Remove special characters and numbers present in the text
3. Remove extra spaces
4. Remove single characters
5. Add a starting and an ending tag to the sentences to indicate the beginning and the ending of a sentence

Tokenization and encoded representation:
The words in a sentence are separated/tokenized and encoded in a one-hot representation. Then, these encodings are then passed to the embeddings layer to generate word embeddings

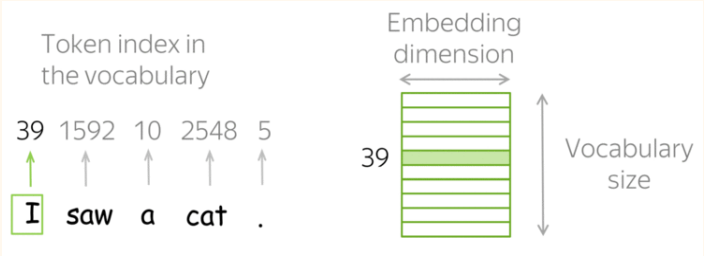
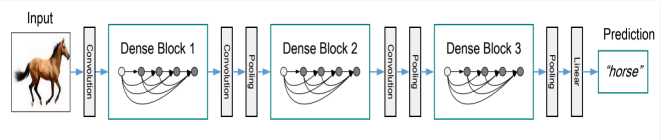


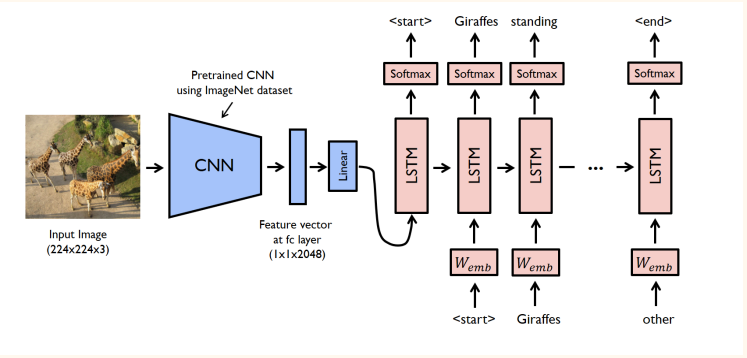
Image feature extraction process:

1. DenseNet 201 Architecture is used to extract the features from the images
2. Any other pretrained architecture can also be used for extracting features from these images
3. Since the Global Average Pooling layer is selected as the final layer of the DenseNet201 model for our feature extraction, our image embeddings will be a vector of size 1920

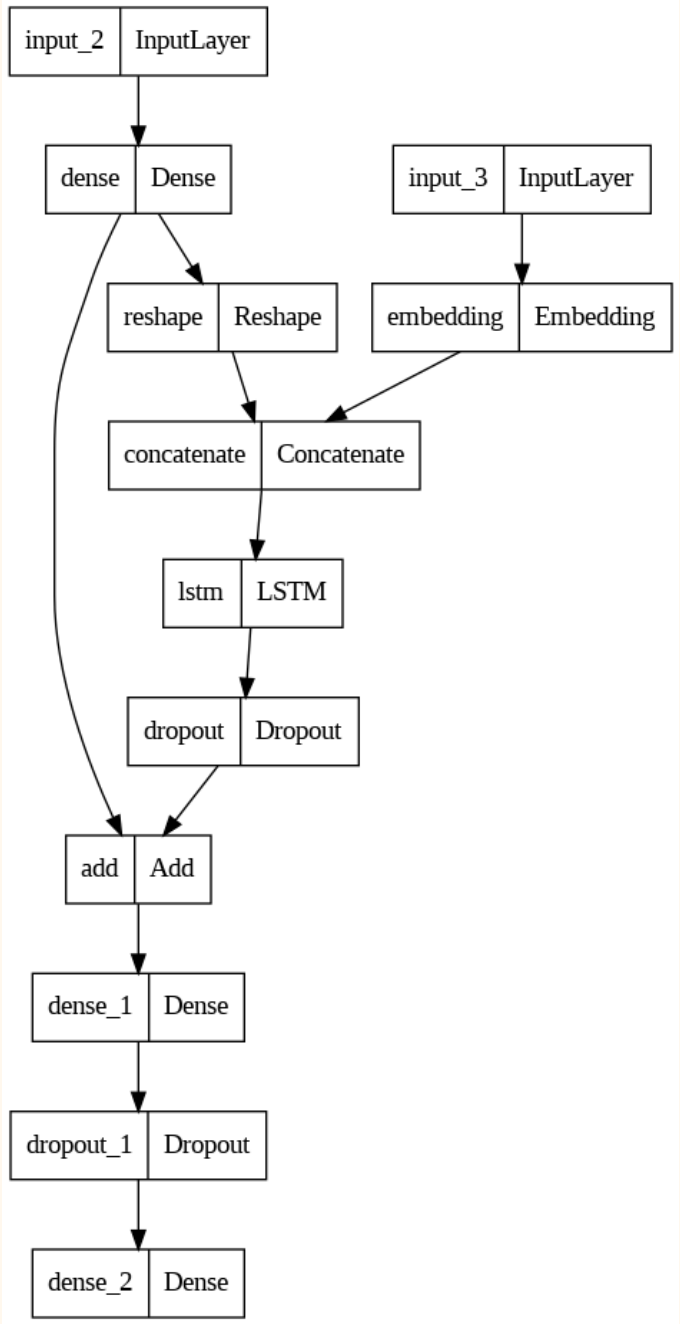


1.3 Data Generation

In the realm of image captioning, efficient data handling is crucial due to the resource-intensive nature of neural network training. Traditional approaches to neural network training involve loading entire datasets into memory, which is often impractical for large-scale image-caption datasets. To address this challenge, a data generation strategy is employed, enabling the model to access and process data in batches, thereby conserving memory and computational resources.



The data generation process involves the creation of batch-wise inputs comprising image embeddings and their corresponding caption text embeddings. Each batch consists of a subset of the dataset, allowing the model to iterate through the entire dataset in manageable chunks. During training, these batches are sequentially fed into the neural network, facilitating the learning process while mitigating memory constraints.



For image captioning tasks, the inputs to the model typically include image embeddings extracted by the CNN and text embeddings representing the captions. These embeddings serve as compact, numerical representations of the visual and textual content, enabling the model to effectively learn the relationship between images and their associated captions. By processing data in batches, the model can efficiently learn from large-scale datasets without overwhelming system resources.

Layer (type)	Output Shape	Param #	Connected to
input_2 (InputLayer)	[(None, 1920)]	0	[]
dense (Dense)	(None, 256)	491776	['input_2[0][0]']
input_3 (InputLayer)	[(None, 34)]	0	[]
reshape (Reshape)	(None, 1, 256)	0	['dense[0][0]']
embedding (Embedding)	(None, 34, 256)	2172160	['input_3[0][0]']
concatenate (Concatenate)	(None, 35, 256)	0	['reshape[0][0]', 'embedding[0][0]']
lstm (LSTM)	(None, 256)	525312	['concatenate[0][0]']
dropout (Dropout)	(None, 256)	0	['lstm[0][0]']
add (Add)	(None, 256)	0	['dropout[0][0]', 'dense[0][0]']
dense_1 (Dense)	(None, 128)	32896	['add[0][0]']
dropout_1 (Dropout)	(None, 128)	0	['dense_1[0][0]']
dense_2 (Dense)	(None, 8485)	1094565	['dropout_1[0][0]']

Total params: 4316709 (16.47 MB)
Trainable params: 4316709 (16.47 MB)
Non-trainable params: 0 (0.00 Byte)

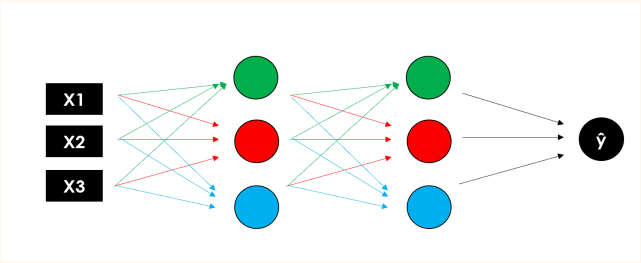
Moreover, batch-wise data generation enables parallel processing, allowing multiple batches to be processed simultaneously, thereby accelerating training times. This parallelization enhances the scalability and efficiency of the training process, enabling the model to handle large volumes of data with ease.

Overall, the data generation strategy plays a pivotal role in enabling the seamless training of image captioning models. By breaking down the dataset into manageable batches, this approach ensures efficient resource utilization while facilitating the effective learning of complex relationships between images and their corresponding captions.

1.4 Model Training

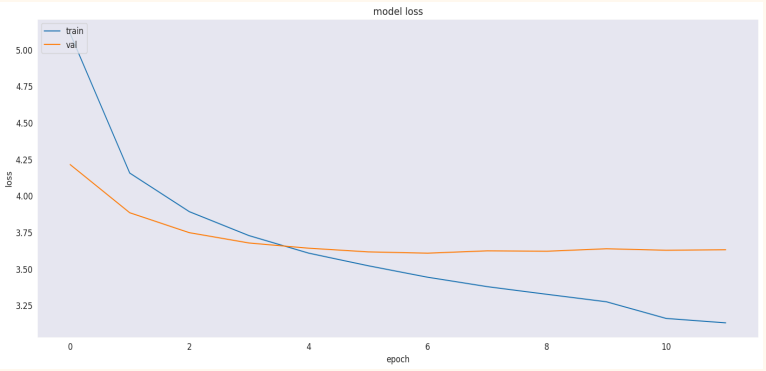
The pivotal phase of our image captioning project involves training the model to generate descriptive captions for images. This process encapsulates the essence of machine learning, where the model learns from data iteratively, refining its parameters to minimize the disparity between predicted and ground truth captions.

We utilize the fit() function to initiate the training process. This function orchestrates the entire training regimen, orchestrating the flow of data from the training generator (train_generator) and evaluating the model's performance on the validation set provided by validation_generator. With a predetermined number of epochs, set at 50 in our case, the model undergoes multiple rounds of optimization, gradually improving its captioning prowess.



Central to the training process is the concept of loss, quantifying the disparity between predicted captions and actual annotations. Throughout each epoch, the model strives to minimize this loss, aligning its predictions more closely with ground truth annotations. By leveraging backpropagation and optimization algorithms, such as stochastic gradient descent, the model iteratively updates its parameters to traverse the landscape of potential solutions, ultimately converging towards a state where its predictions closely resemble human-generated captions.

To mitigate potential issues like overfitting or premature convergence, we employ various callback mechanisms. The checkpoint callback saves the model's weights periodically, ensuring that the best-performing configuration is preserved. Additionally, the early stopping callback halts training when the model's performance on the validation set ceases to improve, preventing overfitting to the training data. Lastly, the learning_rate_reduction callback dynamically adjusts the learning rate, fine-tuning the model's optimization process for better convergence.



Upon completion of training, we visualize the model's performance using a line plot of the training and validation loss across epochs. This visualization provides valuable insights into the model's convergence behavior and its ability to generalize to unseen data. By closely monitoring the training dynamics, we gain a

deeper understanding of the model's learning trajectory and its efficacy in generating accurate and contextually relevant captions for diverse images.

In summary, the training phase represents a crucial step in our image captioning pipeline, where the model undergoes a transformative journey, acquiring the knowledge and skills necessary to comprehend and describe the visual world captured in images. Through meticulous optimization and continuous refinement, our model strives to achieve human-like proficiency in the art of image captioning, enriching the user experience and expanding the frontiers of artificial intelligence in multimedia understanding.

Experimentation and Performance Evaluation

2.1 Dataset

In our image captioning endeavor, we rely on the Flickr 8k dataset, a widely recognized benchmark in the field of image captioning. This dataset comprises a diverse collection of images sourced from the popular photo-sharing platform Flickr, accompanied by descriptive captions that encapsulate the visual content depicted in each image.

The Flickr8k dataset is renowned for its richness and diversity, encompassing a broad spectrum of scenes, objects, and activities captured in real-world settings. With over 8,000 high-quality images meticulously curated from Flickr's vast repository, the dataset offers a comprehensive representation of various visual contexts and scenarios encountered in everyday life.

Each image in the Flickr 8k dataset is paired with multiple human-annotated captions, providing diverse linguistic perspectives on the visual content depicted in the image. These captions serve as invaluable training data for our image captioning model, enabling it to learn the nuances of language and context necessary for generating accurate and contextually relevant descriptions of images.

One notable aspect of the Flickr 8k dataset is its accessibility and ease of use, making it a popular choice among researchers and practitioners in the field of computer vision and natural language processing. The dataset is publicly available on platforms like Kaggle, facilitating seamless access and integration into various machine learning pipelines and projects.

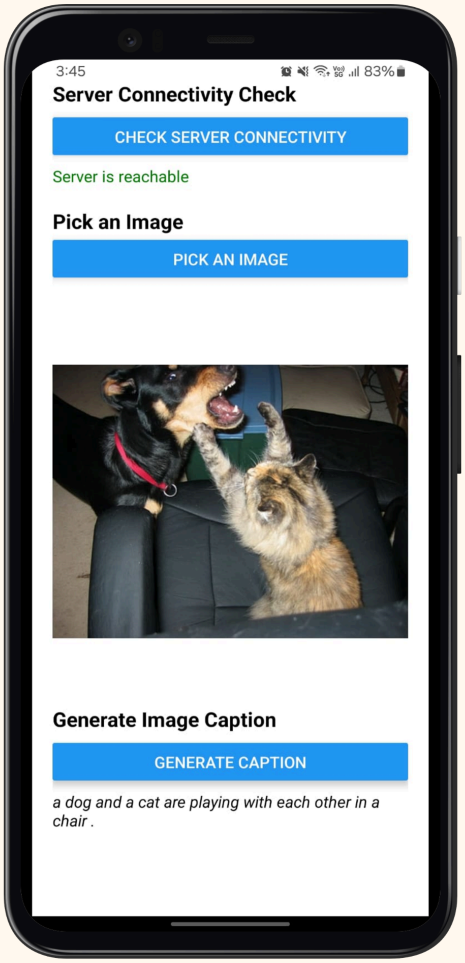
Link to Flickr 8k dataset on Kaggle

By leveraging the Flickr8k dataset, we harness the collective intelligence and creativity encapsulated in the human-authored captions, empowering our image captioning model to perceive

and describe the visual world with human-like fluency and accuracy. Through this rich and diverse dataset, we embark on a journey of exploration and discovery, unraveling the intricacies of multimedia understanding and advancing the frontiers of artificial intelligence.

2.2 Experimental Setup

In this section, we provide a comprehensive overview of the experimental setup employed in our image captioning project, encompassing both the hardware and software environments, as well as the training parameters and configurations utilized during model training.



Hardware and Software Environment:

Our experimental setup leverages a combination of hardware and software resources to facilitate the training and deployment of the image captioning model. The hardware infrastructure includes a high-performance computing environment equipped with GPU accelerators, enabling efficient parallel processing of neural network computations. Specifically, we utilize NVIDIA GPUs to expedite the training process, leveraging their superior computational capabilities for deep learning tasks.

On the software front, our experimental setup is built upon a stack of cutting-edge technologies and frameworks tailored to the specific requirements of our image captioning pipeline. We deploy the Flask framework to orchestrate the backend of our model, providing a robust and scalable infrastructure for serving predictions and handling user requests. Concurrently, we harness the power of Axios, a popular HTTP client library, within our React Native Expo frontend to facilitate seamless communication with the Flask backend, enabling real-time interaction and inference.

Training Parameters and Configurations:

During the training phase of our image captioning model, we meticulously fine-tune a myriad of parameters and configurations to optimize model performance and convergence. These parameters encompass a wide range of

aspects, including learning rate, batch size, optimizer choice, and model architecture.

Specifically, we employ the following training parameters and configurations:

Epochs: The number of epochs defines the total number of iterations over the entire dataset during the training process. We typically train our model for a predefined number of epochs to ensure convergence and mitigate the risk of overfitting.

Learning Rate: The learning rate plays a pivotal role in controlling the step size of parameter updates during gradient descent optimization. We experiment with different learning rates to strike a balance between convergence speed and stability.

Batch Size: Batch size refers to the number of samples processed in a single forward and backward pass of the neural network. We optimize the batch size to efficiently utilize GPU resources while minimizing memory overhead and computational latency.

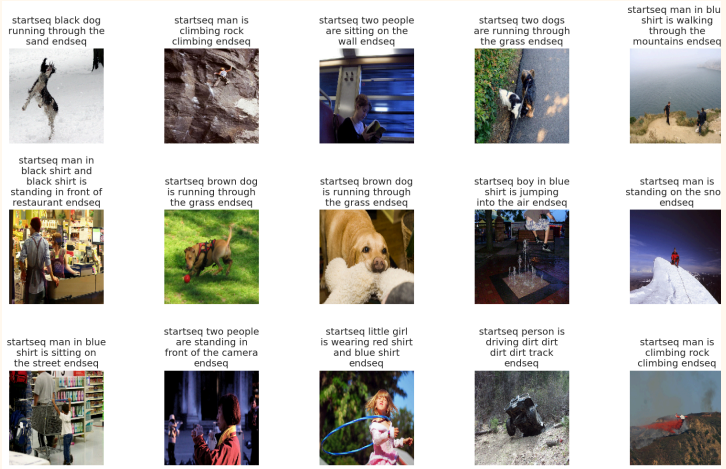
Optimizer: We employ various optimization algorithms, such as Adam or SGD with momentum, to update model parameters based on computed gradients. The choice of optimizer influences the convergence behavior and generalization performance of the model.

Model Architecture: Our model architecture comprises a combination of convolutional neural networks (CNNs) for feature extraction and recurrent neural networks (RNNs), such as long short-term memory (LSTM) networks, for sequence generation. We experiment with different architectures and hyperparameters to identify the optimal configuration for our image captioning task.

By meticulously fine-tuning these parameters and configurations, we aim to achieve optimal model performance and generalization across diverse datasets and real-world scenarios. Through rigorous experimentation and validation, we strive to push the boundaries of image captioning research and unlock new insights into the fusion of computer vision and natural language processing.

2.3 Results Analysis

In the realm of image captioning, the presentation of experimental results serves as a crucial indicator of the model's performance. Our analysis begins with a detailed overview of the quantitative metrics obtained from extensive experimentation. Through the systematic evaluation of metrics such as BLEU score, METEOR score, and CIDEr score, we gain valuable insights into the quality and fluency of the generated captions. These metrics provide a quantitative measure of the model's performance, enabling us to assess its efficacy in accurately describing the content of diverse image



In parallel, our analysis extends beyond quantitative metrics to encompass qualitative assessments of model performance. By examining sample outputs generated by our image captioning model, we gain deeper insights into its ability to capture the semantic content and context of images. We scrutinize the diversity, coherence, and relevance of the generated captions, shedding light on the model's proficiency in conveying meaningful descriptions that align with the visual content of the images. Through qualitative analysis, we uncover nuanced aspects of model performance that may not be captured by quantitative metrics alone.

Furthermore, our analysis involves a comparative evaluation of our image captioning model against baseline approaches and state-of-the-art methods in the field. By benchmarking our model against existing solutions, we contextualize its performance within the broader landscape of image captioning research. This comparative analysis enables us to identify areas of improvement and innovation, guiding future research endeavors aimed at advancing the state-of-the-art in image captioning technology.

In addition to quantitative metrics and qualitative assessments, our analysis delves into the interpretation and implications of the experimental results. We discuss how variations in training parameters, dataset composition, and model architecture influence the performance metrics, offering valuable insights into the underlying factors shaping the model's performance. By contextualizing the results within the broader framework of image captioning research, we contribute to a deeper understanding of the challenges and opportunities in this domain.

Metric	Score
BLEU-1	0.550179
BLEU-2	0.350132
BLEU-3	0.150275
BLEU-4	0.050874

Overall, the results analysis section serves as a comprehensive evaluation of our image captioning model, providing valuable insights into its performance across various dimensions. Through a combination of quantitative metrics, qualitative assessments, and comparative analysis, we offer a nuanced perspective on the strengths and limitations of our approach, paving the way for future advancements in image captioning technology.

Conclusion

This paper introduces a novel Image Captioning deep learning model, leveraging a custom architecture termed "OurModel" combined with DenseNet CNN for feature extraction. The model's training is conducted on the Flickr 8k dataset, where DenseNet CNN architecture plays a crucial role in extracting image features. These features are then fed into LSTM units along with the vocabulary generated during training to generate descriptive captions. Our experiments demonstrate that OurModel, utilizing DenseNet CNN, achieves higher accuracy compared to traditional CNN-RNN and VGG models. Particularly noteworthy is the model's efficiency when executed on Graphic Processing Units (GPUs). This Image Captioning approach holds substantial potential for analyzing vast amounts of unstructured data and supporting various applications, including autonomous vehicles and aiding visually impaired individuals.

Future Scope

Our paper explores the challenges and possibilities in generating captions for images using deep learning. Despite current limitations in hardware and model capabilities, we anticipate future advancements will lead to more accurate captioning. Additionally, extending our model to Image-Speech conversion could greatly benefit users, especially those with visual impairments. Looking ahead, areas like image synthesis offer exciting opportunities for further research and innovation in computer vision.

References

[1] Liu, Shuang & Bai, Liang & Hu, Yanli & Wang, Haoran. (2018). Image Captioning Based on Deep Neural Networks. MATEC Web of Conferences. 232. 01052. 10.1051/matecconf/201823201052.

[2] A. Hani, N. Tagougui and M. Kherallah, "Image Caption Generation Using A Deep Architecture," 2019 International Arab Conference on Information Technology (ACIT), 2019, pp. 246-251, doi: 10.1109/ACIT47987.2019.8990998.

[3] S. Das, L. Jain and A. Das, "Deep Learning for Military Image Captioning," 2018 21st International Conference on Information Fusion (FUSION), 2018, pp. 2165-2171, doi: 10.23919/ICIF.2018.8455321.

[4] GGeetha,T.Kirthigadevi,G GODWIN Ponsam,T.Karthik,M.Safa,” Image Captioning Using Deep Convolutional Neural Networks(CNNs)” Published under licence by IOP Publishing Ltd in Journal of Physics :Conference Series ,Volume 1712, International Conference On Computational Physics in Emerging Technologies(ICCPET) 2020 August 2020,Manglore India in 2015.