Automatic Image Captioning Using Deep Learning and CNN-LSTM Architecture

1st SHREYA S
Artificial Intelligence
National Institute of Technology Karnataka)
Surathkal, India 575025
shreyas.231ai037@nitk.edu.in

2nd DINESH
Artificial Intelligence
National Institute of Technology Karnataka)
Surathkal, India 575025
garbhapudinesh.231ai010@nitk.edu.in

3rd SUNIL
Artificial Intelligence
National Institute of Technology Karnataka)
Surathkal, India 575025
Sunil.231ai038@nitk.edu.in

4th MAHAK
Artificial Intelligence
National Institute of Technology Karnataka)
Surathkal, India 575025
mahak.231ai018@nitk.edu.in

***Abstract*—Image captioning is a challenging task at the intersection of computer vision and natural language processing, aimed at generating accurate textual descriptions for a given image. In this work, a deep learning-based approach is proposed that leverages the combined strengths of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks. The system uses a pre-trained VGG16 model to extract meaningful features from images, which are then passed to an LSTM network that generates grammatically correct and contextually appropriate captions, one word at a time. The model is trained and evaluated using the Flickr8k dataset, which contains images paired with human-annotated captions. The quality of the generated captions is assessed using the BLEU (Bilingual Evaluation Understudy) metric, a standard evaluation technique for comparing machine-generated text against human reference sentences. Experimental results demonstrate that the proposed CNN-LSTM architecture achieves satisfactory performance in generating relevant image descriptions, offering promising potential for applications in automated content annotation, image retrieval, and assistive technologies.**

I. Introduction

The ability of humans to comprehend and describe visual scenes is remarkably intuitive and efficient, yet replicating this capability in machines has been a long-standing challenge in the field of artificial intelligence. With the rapid progress of deep learning techniques, especially in the domains of computer vision and natural language processing, significant strides have been made toward enabling machines to generate meaningful and contextually accurate descriptions of images.

Image captioning is a task that involves generating natural language descriptions for a given image by understanding its visual content. This requires the combination of two distinct but complementary fields: computer vision, for understanding and extracting relevant features from the image, and natural language generation, for constructing coherent and grammatically correct textual descriptions.

Traditional image captioning techniques relied on rule-based approaches, template-based sentence generation, or retrieval of pre-existing captions, which often resulted in generic or inflexible outputs. However, the integration of deep learning has introduced end-to-end models capable of learning both visual features and the sequential dependencies of natural language, resulting in more accurate and context-aware captions.

In recent years, hybrid architectures combining Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks have emerged as a highly effective framework for image captioning tasks. CNNs are leveraged for image feature extraction, capturing spatial hierarchies and object-level details, while LSTMs are employed for sequential caption generation, effectively handling long-range dependencies in language.

The primary objective of this study is to implement an image captioning system that utilizes a CNN-LSTM-based encoder-decoder architecture. The system is trained using the Flickr8k dataset, which provides a diverse set of images with multiple descriptive captions. The model's performance is evaluated using the BLEU (Bilingual Evaluation Understudy) score, a metric widely used for assessing the quality of generated text in comparison to human-generated references.

This approach demonstrates the potential of combining deep learning models to bridge the gap between visual understanding and natural language description, opening new possibilities for applications in content accessibility, image search, autonomous systems, and assistive technologies.

II. . LITERATURE SURVEY

Image captioning is a multidisciplinary problem that merges computer vision and natural language processing. Over the years, several methodologies have been proposed to address this challenge, each aiming to improve the accuracy, fluency, and semantic relevance of the generated captions. Broadly, the methods can be categorized into three main approaches: template-based, retrieval-based, and deep learning-based generation models.

In the template-based approach, predefined sentence structures with placeholders are used. After detecting objects and attributes from the image, these placeholders are filled accordingly. Although this method ensures grammatically correct captions, it lacks diversity and often struggles with complex or unseen image scenarios.

Retrieval-based techniques select captions directly from a set of candidate captions in the training dataset. Given an input image, the model retrieves the most visually similar image and uses its associated captions. While retrieval-based systems can produce human-like captions, they fail to generate novel descriptions for unseen image contexts.

The recent and most promising advancements stem from deep learning-based generation methods, particularly the combination of Convolutional Neural Networks (CNNs) for image feature extraction and Recurrent Neural Networks (RNNs) or Long Short-Term Memory (LSTM) networks for sequence generation. This encoder-decoder architecture enables the model to understand visual content and generate coherent, contextually meaningful natural language descriptions.

Furthermore, advanced captioning models have integrated attention mechanisms, allowing the system to focus on different regions of an image when generating each word, thus improving both the relevance and precision of the output.

Several noteworthy contributions in this field include:

Hossain et al. [1] provided a comprehensive survey of deep learning techniques applied to image captioning, highlighting both the strengths and limitations of state-of-the-art models. You et al. [2] introduced the concept of semantic attention to guide the caption generation process, allowing the model to focus on the most salient visual features. Park et al. [3] proposed Context Sequence Memory Networks (CSMN), which enhance the model's ability to store and utilize long-term dependencies in both visual and textual data. Johnson et al. [4] presented DenseCap, a model for dense captioning that not only generates image-level captions but also localizes descriptive phrases to specific regions of an image. Wang et al. [7] and Shi et al. [12] explored methods to improve the quality of generated captions by incorporating better utilization of existing annotated datasets. These works have laid the foundation for modern image captioning models, leading to the development of systems that can produce more semantically rich and syntactically accurate captions. The current research continues to refine these models by exploring attention mechanisms, alternative feature extractors, and transformer-based architectures.

III. PROBLEM STATEMENT

The problem is to automatically generate meaningful and grammatically correct captions for images using artificial intelligence. I planned to solve it by designing a deep learning-based image captioning system that combines Convolutional Neural Networks (CNN) for visual feature extraction and Long Short-Term Memory (LSTM) networks for generating descriptive, context-aware sentences.

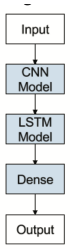

Fig. 1. CNN-LSTM model structure

Fig. 1. CNN-LSTM model structure

IV. METHODOLOGY

The image captioning system first processes and encodes images using a CNN to obtain feature vectors. These features are then passed to an LSTM-based language model which predicts word sequences describing the image. The model is trained on paired datasets of images and their textual descriptions to learn the mapping between visual features and natural language.

- Keras: Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano.One of the key strengths of Keras is its simplicity and ease of use. Keras also provides a number of pre-trained models, making it easy to get started with transfer learning and fine-tuning existing models.
- Relevant Python libraries like NumPy, Pickle- for serializing and deserializing Python objects, tqdm-to get progress feedback through the progress bar widget for long-running tasks or iterations over a large datasets.
- Tokenizer: Tokenization is the process of breaking down a piece of text into smaller units, known as tokens. Tokens can be words, phrases, symbols, or other elements of the text, depending on the specific use case. Tokenization is a fundamental step in many natural language processing (NLP) tasks, including text classification, sentiment analysis, and information retrieval.Tokenization is important because it allows us to represent text in a structured format that can be easily processed by computers.

A. Pre-processing the Image

In our project, we are using a pre-trained model named VGG16 from the Visual Geometry Group for image recognition. This model is a available within the Keras library, and thus it will not require additional installation or setup. For this project, the image features are extracted by resizing the images to a size of 224*224 pixels. The extraction process is performed on the image just before the last layer of the classification model.
This location is chosen because it is used for predicting the classification of a photo, but since it is not required to classify the images, we exclude the last layer during the feature extraction process.

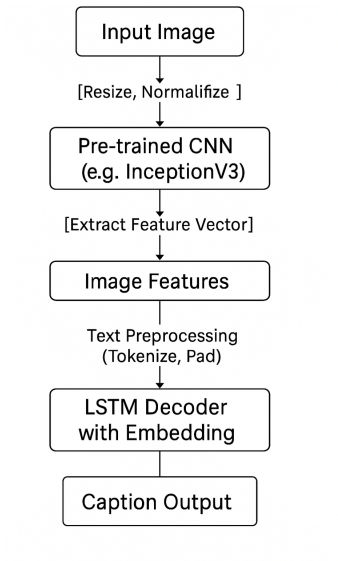

Fig. 2. WORKFLOW

block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
fc1 (Dense)	(None, 4096)	102764544
fc2 (Dense)	(None, 4096)	16781312

Total params: 134,260,544
Trainable params: 134,260,544
Non-trainable params: 0

None

Fig. 6. VGG16 model parameters

Fig. 3. VGG16 model parameters

B. Creating vocabulary for the image:

Before using text data in a machine learning or deep learning model, it is necessary to clean and prepare it for the model. This process includes splitting the text into individual words, handling issues with punctuation and case sensitivity. Additionally, since computers do not understand English words, we need to represent them with numbers. This is done by creating a vocabulary and mapping each word to a unique index value, and then encoding each word into a fixed-sized vector. Only after this process, the text becomes readable by the machine and can be used to generate captions for images. We plan to reduce the size of our vocabulary by processing the text in the following sequence through cleaning. In order to accomplish the goals of reducing the size of vocabulary, we have outlined and defined five functions below.

- Retrieving the data..
- Establishing a mapping between images and their descriptions using a dictionary.
- Purifying the descriptions by eliminating punctuation marks, transforming them to lowercase letters, and removing any words containing numbers.
- Create a list of all the distinct words found in the descriptions and make a vocabulary out of them.
- Making a document to keep all the captions saved.

In order to establish a vocabulary for the project, all the unique words from the training dataset are tokenized. This process results in 8763 unique words being defined as the vocabulary.

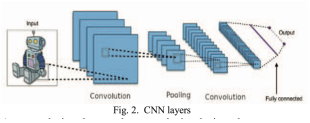

Fig. 2. CNN layers

Fig. 4. CNN layers

C. Training the model:

The dataset includes a file named ”trainimages.txt”, which contains a compiled list of 8,000 image names that are used during the training phase of the project. The first step in the training process involves loading the image features, which were previously extracted using the pre-trained CNN model. Once the features are loaded, the training images are divided into smaller chunks known as batches. These batches are then used to generate corresponding input and output sequences, allowing the model to progressively learn the relationship between image features and their respective textual descriptions. The model is trained using these sequences over 20 epochs, enabling it to gradually optimize its predictions and improve caption generation performance.

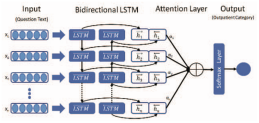

Fig. 3. LSTM structure

Fig. 5. CNN-LSTM Model Architecture for Image Captioning

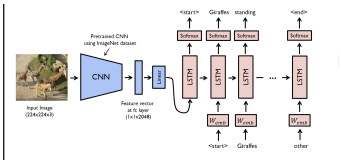

Fig. 6. CNN-LSTM model

D. Model evaluation :

BLEU (Bilingual Evaluation Understudy) is a widely used method for measuring the similarity between a generated sentence and a reference sentence. It is primarily applied to evaluate the performance of machine translation systems, but it is also well-suited for assessing image captioning models. The BLEU score ranges from 0.0 to 1.0, where 1.0 represents a perfect match between the generated caption and the reference caption, while 0.0 indicates a complete mismatch.

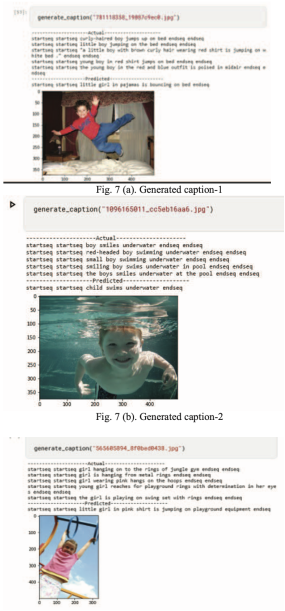

Fig. 7 (a). Generated caption-1

Fig. 7 (b). Generated caption-2

Fig. 7 (c). Generated caption-3

Fig. 7. Generated caption-3

In this project, the BLEU score is used to evaluate how accurately the model generates captions for images. The generated captions are compared against the corresponding human-annotated reference captions, and the similarity between the two sets is measured. The BLEU score is calculated across the entire set of test captions, providing a quantitative summary of how well the model has learned to produce relevant, coherent, and meaningful descriptions that align with human expectations.

E. Overfitting Prevention Using Dropout Regularization

To prevent the model from overfitting during the training process, a dropout layer with a rate of 0.5 was applied. Dropout is a regularization technique that helps improve the model's ability to generalize by randomly deactivating 50percentage of the neurons in the network during each training iteration. This forces the model to avoid becoming too dependent on specific neurons and encourages it to learn more robust and diverse feature representations. By using dropout, the chances of the model memorizing the training data are reduced, which leads to improved performance on unseen images and results in more accurate and contextually relevant caption generation.

V. RESULTS AND ANALYSIS

The proposed CNN-LSTM model was trained and evaluated using the Flickr8k dataset, which contains 8,091 images, each annotated with five human-generated captions. The model's performance was assessed using the BLEU (Bilingual Evaluation Understudy) score — a standard metric for evaluating machine-generated text against reference sentences.

After the training phase, the model was able to generate descriptive captions for unseen test images. The quality of the generated captions was evaluated based on BLEU-1 (single-word match) and BLEU-2 (word-pair match) scores.

The obtained results are as follows:

- BLEU-1 Score: 0.5699
- BLEU-2 Score: 0.3766 These scores indicate that the model achieved approximately 57 percentage single-word matching accuracy and 37 percentage two-word phrase accuracy against the reference human-annotated captions, which is considered satisfactory for baseline image captioning models.

  Sample outputs for selected images are shown in Fig. 7, demonstrating the model's ability to generate relevant and meaningful descriptions. The performance was also observed to be sensitive to the number of training epochs, where overfitting was reduced using a dropout rate of 0.5 during training.

VI. CONCLUSION

This work, an image captioning system was successfully implemented using a hybrid deep learning architecture combining a Convolutional Neural Network (VGG16) for image feature extraction and an LSTM network for natural language caption generation.

The experimental evaluation using the Flickr8k dataset demonstrated that the model can generate contextually accurate and syntactically valid captions. The system achieved a BLEU-1 score of 0.5699 and a BLEU-2 score of 0.3766, showing its ability to align closely with human-written descriptions.

While the proposed model yields satisfactory results, there is potential for further enhancement. Future work can focus on integrating attention mechanisms to allow the model to dynamically focus on different image regions during caption generation, which is expected to improve both the semantic richness and the relevance of the generated descriptions.

REFERENCES

[1] M. Z. Hossain, F. Sohel, M. F. Shiratuddin, and H. Laga, "A Comprehensive Survey of Deep Learning for Image Captioning," arXiv preprint, arXiv:1810.04020v2, 2018.

[2] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image Captioning with Semantic Attention," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 4651–4659.

[3] C. Park, B. Kim, and G. Kim, "Attend to You: Personalized Image Captioning With Context Sequence Memory Networks," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 895–903.

[4] J. Johnson, A. Karpathy, and L. Fei-Fei, "DenseCap: Fully Convolutional Localization Networks for Dense Captioning," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 4565–4574.

[5] J. Aneja, A. Deshpande, and A. G. Schwing, "Convolutional Image Captioning," in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 5561–5570.

[6] G. Srivastava and R. Srivastava, "A Survey on Automatic Image Captioning," in Communications in Computer and Information Science (CCIS), vol. 834, Springer, 2018.

[7] H. Wang, Y. Zhang, and X. Yu, "An Overview of Image Caption Generation Methods," Computational Intelligence and Neuroscience, vol. 2020, Article ID 3062706, 13 pages, 2020.

[8] Sharma, N. Ding, S. Goodman, and R. Soricut, "Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning," in Proc. 56th Annual Meeting of the Association for Computational Linguistics (ACL), 2018, pp. 2556–2565.

[9] G. Sairam, M. Mandha, P. Prashanth, and P. Swetha, ”Image Captioning using CNN and LSTM,” in Proc. 4th Smart Cities Symposium (SCS), Bahrain, 2021, pp. 274–277.
[10] G. Sairam, M. Mandha, P. Prashanth, and P. Swetha, ”Image Captioning using CNN and LSTM,” in Proc. 4th Smart Cities Symposium (SCS), Bahrain, 2021, pp. 274–277.
[11] V. Agrawal, S. Dhekane, N. Tuniya, and V. Vyas, ”Image Caption Generator Using Attention Mechanism,” in Proc. 12th Int. Conf. on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2021.
[12] TensorFlow, [Online]. Available: https://www.tensorflow.org