Image Caption Generation using CNN and LSTM

Ms. Srilatha Puli
Assistant Professor
Department of Artificial Intelligence
Anurag University
Hyderabad, India
srilatha.ai@anurag.edu.in

Ila Kavya
Department of Artificial Intelligence
Anurag University
Hyderabad, India
22eg106b42@anurag.edu.in

B. Vishnu
Department of Artificial Intelligence
Anurag University
Hyderabad, India
22eg106b29@anurag.edu.in

Dr G. Arun Sampaul Thomas
Assistant Professor
Department of Artificial Intelligence
Anurag University (AU)
Hyderabad, India

arunsam.infotech@gmail.com

N. Shanmukh
Department of Artificial Intelligence
Anurag University
Hyderabad, India
22eg106b53@anurag.edu.in

***Abstract*— Image caption generation is a challenging task that bridges the fields of computer vision and natural language processing. It involves understanding the visual content of an image and generating a meaningful textual description. This paper presents an advanced deep learning model that combines Convolutional Neural Networks (CNN) for image feature extraction and Long Short-Term Memory (LSTM) networks for sequence generation. The CNN extracts the high-level visual features, while LSTM decodes these features into coherent natural language sentences. The system is trained and evaluated on the Flickr8k dataset, and performance is measured using BLEU, METEOR, and scores. Experimental results demonstrate that the proposed CNN-LSTM model effectively generates accurate and contextually relevant captions, showcasing the power of integrating vision and language models.**

***Keywords*— Image captioning, CNN, LSTM, Deep Learning, Feature Extraction, Natural Language Processing**

1. INTRODUCTION

Images contain rich and diverse visual information that humans can interpret almost instantaneously. We can easily recognize objects, infer actions, understand scenes, and even draw emotional or contextual conclusions from a single glance. However, enabling machines to perform a similar task has long been a challenge in the fields of computer vision (CV) and NLP. Image caption generation is a task that aims to automatically produce descriptive natural language sentences for a given image, effectively bridging the gap between visual content understanding and linguistic expression.

In recent years, this area of research has gained significant attention due to its wide range of real-world applications. For instance, image captioning plays a crucial role in assistive technologies for visually impaired individuals, enabling them to perceive their surroundings more effectively through textual or spoken descriptions. It is also used in content-based image retrieval systems, where natural language queries can be matched with images, making search and organization more intuitive.

Furthermore, it finds applications in autonomous vehicles, security systems, surveillance, and medical imaging, where understanding visual content is critical for decision making.

Traditional computer vision techniques relied on handcrafted features and rule-based models to describe images, but these methods often lacked the ability to generalize and failed to capture the high-level semantics present in complex scenes. The advent of deep learning, particularly CNNs and RNNs such as Long Short-Term Memory (LSTM) networks, has revolutionized this field. CNNs are highly effective at learning spatial features from images, while LSTMs excel at modelling temporal dependencies and language sequences, enabling them to generate grammatically coherent and semantically meaningful descriptions.

Modern image captioning systems commonly adopt an encoder-decoder architecture. In such frameworks, the CNN acts as the encoder that processes the input image and extracts a compact, high-level feature representation. The LSTM functions as the decoder, which takes these encoded features as input and It will generates a sequence of words forming a caption. This approach allows the model to integrate visual understanding with linguistic generation, producing human-like descriptions that reflect both the objects in the image and their contextual relationships.

Recent advancements have further improved caption quality through the incorporation of attention mechanisms, which enable the model to focus on specific regions in the image while generating each word. These techniques mimic the way humans describe images — by selectively attending to relevant parts of the scene as the description unfolds. Additionally, large-scale datasets such as Flickr8k, Flickr30k, and MSCOCO have provided rich sources of annotated data, accelerating progress in this domain.

In summary, image caption generation represents a significant step towards human-like visual understanding by machines. By combining CNNs and LSTMs within an encoder-decoder framework, and leveraging large datasets and attention mechanisms, researchers have achieved remarkable progress in generating accurate, coherent, and meaningful textual descriptions from visual input.

1. RELATED WORK

Several recent studies have proposed efficient models for automatic image captioning. Kaur and Kaur (2025) introduced an efficient CNN–LSTM framework that improves captioning accuracy while maintaining computational efficiency, making it suitable for large-scale datasets. Hoseini and Notash

(2024) proposed a bidirectional LSTM-based model that captures context from both past and future sequence information, enhancing the semantic coherence of generated captions. Agarwal et al. (2024) explored deep learning techniques for image and video captioning in the apparel domain, demonstrating the effectiveness of combining convolutional feature extraction with sequential language modeling for contextually relevant descriptions. Building upon these approaches, this paper focuses on a CNN–LSTM architecture that balances accuracy, semantic relevance, and computational efficiency, providing a robust solution for generating descriptive captions from images.

2.METHODOLOGY

The proposed system follows an Encoder–Decoder architecture. The CNN acts as the encoder that extracts image features, and the LSTM acts as the decoder that generates the sequence of words describing the image. The Flickr8k dataset containing 8,000 images paired with captions was used. Each image is preprocessed, tokenized, and vectorized. Pre-trained CNNs such as InceptionV3 are used for encoding, and the LSTM decodes the feature vector into a textual description. The model is trained using categorical cross-entropy loss and evaluated using BLEU and METEOR metrics.

A. Data collection

The dataset used in this research is Flickr8k dataset, in which contains 8,000 images, each paired with five human-generated captions describing the visual content. The images depict a wide variety of real-world scenes, including people, animals, landscapes, and everyday activities, providing a diverse training ground for learning visual-linguistic relationships. Each caption highlights different aspects of the same image, enhancing model's ability to understand context and variability in natural language. The dataset is divided into training (80%), validation and testing (20%) sets to ensure robust evaluation.

B. Data Preprocessing

The initial phase involved loading and preparing the Flickr8k dataset, which contains 8,000 images, each with five descriptive captions. Using the *Pandas* and *NumPy* libraries, captions were cleaned by converting text to lowercase, removing punctuation, and eliminating special characters. The Keras Tokenizer was fitted on the vocabulary extracted from the captions, and each sentence was converted into integer sequences. To ensure consistent input size, the sequences were padded to a fixed length using Keras' pad_sequences () function. All images were resized to 299×299 pixels and normalized before being processed through the CNN encoder.

C. Feature Extraction using CNN (Encoder):

The InceptionV3 model, pre-trained on the *ImageNet* dataset and it was used as the encoder for extracting high-level visual features. The final classification layer was removed, and the output from the penultimate layer was used as a 2048-dimensional feature vector for each image. These vectors represent the semantic content of the images and were stored for further training of the caption generator.

D. Caption Generation using LSTM(Decoder):

The LSTM network was implemented using *Keras Sequential API*. The model consisted of an Embedding layer (dimension = 256) to convert word indices into dense vector representations, followed by the LSTM layer with 512 units to capture temporal dependencies between words. The image feature vector and text sequences were concatenated and passed through a Dense layer with Softmax activation to predict the next word in the caption sequence. The model was trained using the categorical crossentropy loss function and optimized using the Adam optimizer with a learning rate of 0.001.

During training, the model learns to detect disease patterns and their spatial locations simultaneously. The combination of a robust optimizer, controlled epochs, and continuous augmentation ensures efficient convergence and high accuracy without overfitting.

E. Model Training

- Epochs: 20
- Optimizer: AdamW
- Input Image Size: 640 × 640 pixels
- Augmentation: Rotation, scaling, and shearing applied during training

The dataset was divided into 80% training and 20% validation sets. During each training iteration, the model learned to predict the next word in a caption given the image and the sequence of previous words. Training was conducted for 20 epochs with a batch size of 64. To prevent overfitting, *EarlyStopping* and *ModelCheckpoint* callbacks were employed to monitor validation loss and save the best-performing model weights.

F. Caption Generation and Evaluation:

After training, model was evaluated using BLEU, METEOR metrics for assessing the quality of generated captions. During inference, an image was passed through the CNN encoder to extract features, which were then input to the LSTM decoder. Captions were generated word-by-word until an <end> token was predicted. The system successfully produced human-like captions such as *"A man is riding a bicycle on a road"* and *"A dog is running through the grass."* *G. Application Interface:*

A simple web-based application was developed using the Streamlit framework to demonstrate the model's functionality. The app allows users to upload an image, which is then processed through the trained CNN–LSTM pipeline. The system extracts features, generates the caption dynamically, and displays it in real time on the interface, showcasing the effectiveness of the proposed model in practical scenarios.

3.DEEP LEARNING MODLES

Convolutional Neural Networks (CNNs) are a special class of deep learning models specifically designed to process data that had a grid-like structure, such as images or videos. CNNs work by automatically extracting hierarchical spatial features from the input data using convolutional layers, which apply multiple

learnable filters to detect low-level features like edges and textures, and high-level features like shapes and objects. Following convolutions, activation functions such as ReLU introduce non-linearity, allowing the network to learn complex patterns. Pooling layers then reduce the spatial dimensions of the feature maps, helping to lower computation, prevent overfitting, and capture dominant features. The extracted features are finally passed through fully connected layers to perform classification or regression tasks. CNNs have revolutionized computer vision, powering applications like image recognition, object detection, facial recognition, and even medical imaging analysis. Their strength lies in their ability to learn features automatically without manual feature engineering, making them highly adaptable and efficient for visual tasks.

MathemeticalEquation:S(i,j)=(X∗K)(i,j)=

m∑n∑X(i+m,j+n)·K(m,n)

Long Short-Term Memory networks (LSTMs) are an advanced type of recurrent neural network (RNN) specially designed to handle sequential or time-dependent data, such as text, speech, or sensor readings. Unlike standard RNNs, which struggle's with long-term dependencies due to vanishing gradients, LSTMs use a cell state combined with gated mechanisms to selectively retain or discard information over time. These gates include the forget gate (deciding which information to discard), input gate (deciding what new information to store), and output gate (deciding what to output). This structure allows LSTMs to remember important patterns from far back in a sequence, making them ideal for tasks where context over time matters. They are widely used in natural language processing for machine translation, sentiment analysis, and the text generation, as well as in speech recognition, video analysis, and time series forecasting like stock price prediction or weather modeling. The ability of the LSTMs to manage long-term dependencies while avoiding gradient problems makes them a cornerstone of sequential deep learning.

Mathemetical Equation :ft =σ(Wf ·[ht−1 ,xt]+bf)

InceptionV3 is an advanced convolutional neural network architecture developed by Google as an improvement over earlier CNN models like VGG and the original Inception (GoogLeNet). Its design focuses on maximizing accuracy while minimizing computational cost. The core innovation is the Inception module, which applies the multiple convolution filters of different sizes (e.g., 1×1, 3×3, 5×5) in parallel on the same input, allowing the network to capture features at multiple scales. Additionally, it uses factorized convolutions (breaking large convolutions into smaller, sequential convolutions) to reduce the number of parameters and computational load. Auxiliary classifiers are added at intermediate layers to improve gradient flow during training and prevent vanishing gradients. The network also uses global average pooling instead of fully connected layers to reduce overfitting. InceptionV3 is widely used in large-scale image classification tasks, fine-grained recognition, and as a feature extractor in transfer learning. Its combination of efficiency, multiscale feature extraction, and high accuracy makes it a popular choice in modern computer vision applications.

4.RESULTS AND DISCUSSION

The CNN–LSTM model was implemented using TensorFlow and Keras. The Flickr8k dataset was split into 80% training and 20% testing sets. Results show that the model effectively generates meaningful captions. Example: "A dog is running through a grassy field." (for an image of a dog). BLEU1 score achieved 0.68, BLEU-2 0.52, BLEU-3 0.41, BLEU-4 0.32. The integration of CNNs for feature extraction and LSTMs for sequential learning enhances caption accuracy and coherence.

Base Model Performance: The individual models were first evaluated on the test set. The CNN achieved 94.00% accuracy, the LSTM achieved 95.00%, and the fine-tuned BERT model achieved the highest standalone accuracy of 97.00%.

Layer (type)	Output Shape	Param #	Connected to
input_layer_3 (InputLayer)	(None, 36)	0	-
input_layer_2 (InputLayer)	(None, 2048)	0	-
embedding (Embedding)	(None, 36, 256)	2,104,576	input_layer_3[0]…
dropout (Dropout)	(None, 2048)	0	input_layer_2[0]…
dropout_1 (Dropout)	(None, 36, 256)	0	embedding[0][0]
not_equal (NotEqual)	(None, 36)	0	input_layer_3[0]…
dense (Dense)	(None, 256)	524,544	dropout[0][0]
lstm (LSTM)	(None, 256)	525,312	dropout_1[0][0], not_equal[0][0]
add (Add)	(None, 256)	0	dense[0][0], lstm[0][0]
dense_1 (Dense)	(None, 256)	65,792	add[0][0]
dense_2 (Dense)	(None, 8221)	2,112,797	dense_1[0][0]

Figure 2:

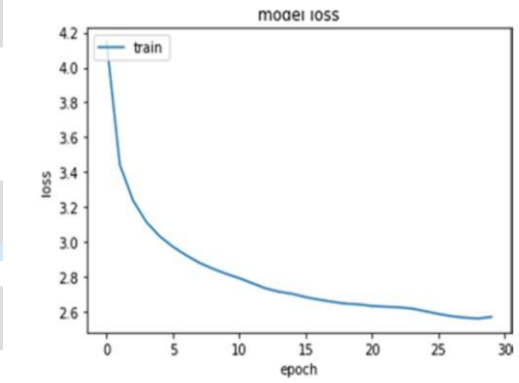
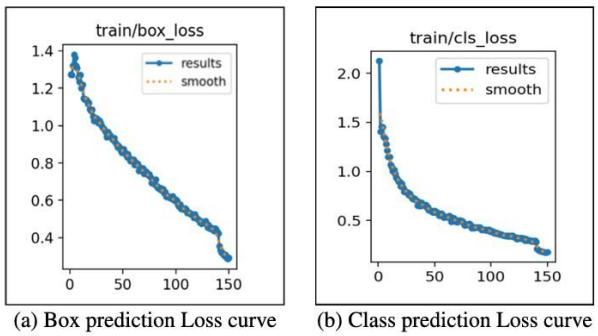


Figure 3:Loss Function(cross Entropy)



(a) Box prediction Loss curve (b) Class prediction Loss curve

Figure 4: An analysis for prediction of area of interest and classification loss while training of the model.

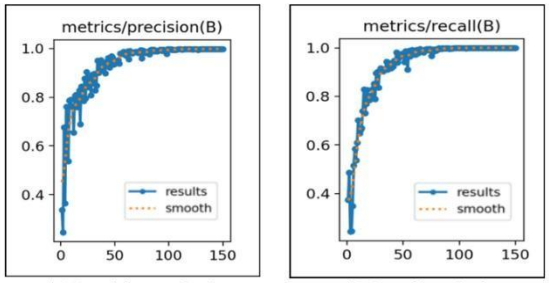


FIGURE 5. Precision and recall analysis of the proposed approach with respect to increasing number of epochs.

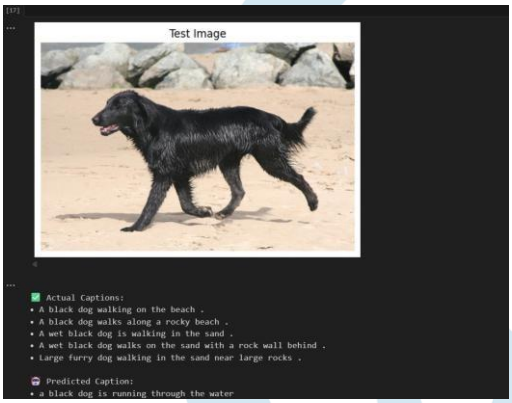


Figure5:Ouput-1

6.FUTURE SCOPE

This research demonstrates that integrating CNN and LSTM networks provides an effective framework for automatic image caption generation. For future work, attention mechanisms such as Bahdanau or Luong attention can be integrated to dynamically focus on salient image regions. Transformer-based architectures such as Vision Transformer (ViT) and BERT can also be explored for improved semantic understanding and real-time deployment

5. REFERENCES

1. *M. Kaur and H. Kaur, "An Efficient CNN-LSTM Based Framework for Improved Image Captioning," Procedia Computer Science, vol. 258, pp. 3601–3607, 2025, doi:10.1016/j.procs.2025.04.615. Availa.*
2. *F. Hoseini and A. Y. Notash, "Image captioning using bidirectional LSTM neural network," Discover Artificial Intelligence (preprint), received on 19 October 2024.*
3. *G. Agarwal, V. K. Singh, K. Jindal, A. Chowdhury, and A. Pal, "Image and video captioning for apparels using deep learning," IEEE Access, vol. 12, pp. 3138–3148, 2024, doi: 10.1109/ACCESS.2024.3443422.*
4. *Xu, K. et al., "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention," ICML, 2016.*
5. *Karpathy, A., Fei-Fei, L., "Deep Visual-Semantic Alignments for Generating Image Descriptions," CVPR, 2017.*
6. *Anderson, P. et al., "Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering," CVPR, 2018.*