Journal of Neonatal Surgery
ISSN(Online): 2226-0439
Vol. 14, Issue 13s (2025)
https://www.jneonatalsurg.com

OPEN ACCESS

Image Caption Detector Using LSTM and CNN

Atyam Mehermani Meghana[1], B. Sekharbabu[2]

[1]M.Tech, Dept Of Cse, Koneru Lakshmaiah Educational Foundation, Vaddeswaram,Guntur,A.P
Email ID: Meghana.Atyam27@Gmail.Com
[2]Associate Professor, Dept Of Cse, Koneru Lakshmaiah Educational Foundation, Vaddeswaram,Guntur,A.P
Email ID: Sekharbabu@Kluniversity.In

Cite this paper as: Atyam Mehermani Meghana, B. Sekharbabu, (2025) Image Caption Detector Using LSTM and CNN. *Journal of Neonatal Surgery*, 14 (13s), 102-112.

ABSTRACT
In computer vision, the general quality is determined by how well the image is comprehensible. Image Captioning is a concept where many models are proposed for a better understanding of the image. In recent times, this technology has been used in many fields like recommendation systems, News channels, Accident detection, and many more. The existing deep learning technique is the Bag of Words model, Spatial pyramid matching and Markov models are used for the automation caption generation but have difficulty in capturing content. So, the proposed methodology is LSTM and CNN are used for effective image representation and have end-to-end learning of the image. VGG16 is used for feature extraction as it has the benefit of hierarchical feature representation. This paper uses the Flickr8K dataset and contains the images along with the five different captions concerning each image. After training, the predictions are made on the model and then evaluated using the metrics BLEU 1, and BLEU 2 which measure the overlapping of the words between generated captions and reference captions and achieves better results than existing models.

Keywords: *LSTM, CNN, VGG16, Flickr8K, BLEU.*

1. INTRODUCTION

Image Captioning is nothing but creating a meaningful sentence for the image based on the objects present. The quality of the image depends on how well you can understand the image. So, the descriptions provided by the model can also be affected based on the quality of the image. Usage of this Image captioning in search engines can also increase the search capacity and provide the most information to the users. It is used in Robotics for understanding a scene while scanning by IOT devices. Image Captioning can used in different industries like social media and news reading. There are many ways to develop a model for generating captions for images. There are some basic steps for the generation of captions. Firstly, Feature Extraction is for extracting the objects in the image and includes background details. Secondly, Sequence modeling is to produce words in a sentence with a correct meaning.

1.1. Image Captioning

Image Captioning is a combination of different fields like Computer Vision and also Deep Learning. Image Captioning consists of both traditional techniques and modern techniques. In the below figure 1, the classification of techniques is there.

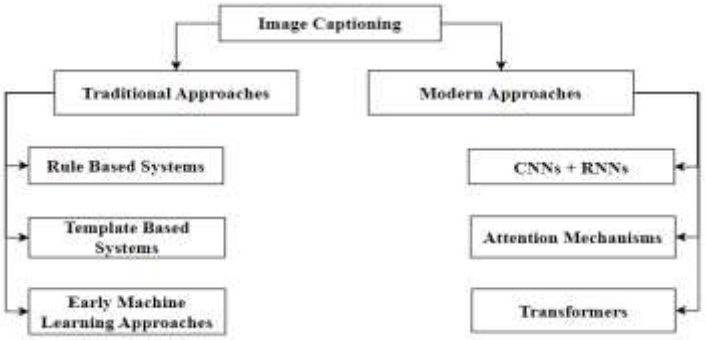

Figure 1: Image Captioning Techniques

1.2. Traditional Approaches

The traditional techniques for image captioning are Rule-Based Systems, Template-based Methods, and Early Machine Learning Approaches. The Rule-Based system uses the rules that are predefined. And those rules help the model in generating the new captions. The Template-Based Method uses sentences that are predefined for finding the objects. Then the model uses those details for the generation of captions. It is very similar to the Rule-Based Systems. The Early Machine Learning Approaches mostly used the SVMs (Support Vector Machines) and Decision Trees. It will extract and classify features and then generate captions.

1.3. Modern Approaches

The Modern approaches consist of CNNs + RNNs, Attention Mechanisms, and Transformers. The CNNs + RNNs are a combination of Encoder and Decoder. The Encoder is used to work on input. The decoder works to generate the captions for the input. The Attention Mechanisms mostly try to mimic the visuals of humans. It focuses on every part of the image to produce the most correct semantic description. The self-attention mechanisms are used by transformer model to generate the captions instead of depending on any RNN model.

1.4. Encoder-Decoder Model

When compared to traditional approaches, modern approaches are much better. The limitations are extensive manual feature engineering, limited generalization, and scalability issues. So, the Encoder-Decoder model is best and reduces those disadvantages. Below figure 2, shows the architecture of the Encoder-Decoder model. The Encoder-Decoder model is used in this paper. The CNN is used in the Encoder model and LSTM is used in the Decoder model. The image is passed through the encoder and features are extracted from the images. The decoder will produce the sequence of words from a sentence and give it as output.

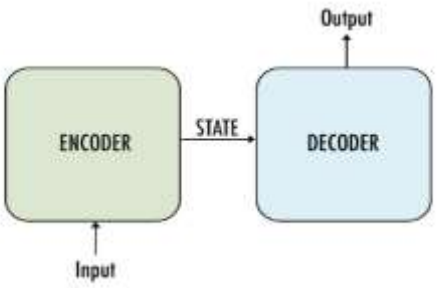

Figure 2: Encoder Decoder Model Architecture

2. DISCUSSION

L. Ramos et al. [1] uses the ConvNeXt to extract features and it is also combined with the LSTM block and also a visual attention module. The models are tested in different scenarios like ConveNeXt versions, different learning rates and teacher force algorithm inclusion and exclusion. This paper used the MSCOCO dataset and gain the best accuracy when the model training is done by less learning rate and exclusion of teacher forcing algorithm. The model has achieved the best BLEU metrics with low convergence and high accuracy of 77.797 and loss of 2.783. The scope of data is limited and works differently on different domains of the data and unique characters are neglected while producing the caption for any domain specific data. The future scope is to achieve better results by adding the hyper parameters and also using the cutting-edge technologies and machines for boosting accuracy.

M. A. Arasi et al. [2] uses the Sparrow search algorithm for automatic image captioning. It also uses the MobileNetv2 model for feature extraction process along with SSA. After that the Attention mechanism along with LSTM for image captioning process and then FFO (fruit fly optimization for hyperparameter tuning process. This model is performed on 3 different datasets that is MS COCO, Flickr8k, Flickr30k. The proposed model has shown the better results in performance measures of image captioning. In future, it can be enhanced using the ensemble fusion process and it can also be tested on large datasets.

S. Amirian et al. [3] reviews about the image and video frame captioning. It describes about the different techniques that used in the both video and image to generate captions. Generally, GANs and CNN are used in the image captioning and CNN+LSTM and LSTM+GAN are used in video captioning. It also reviews the different datasets MS COCO, Flickr datasets for image and Charades, MSVD are some of them for video captioning. The evaluation measures for image and video captioning are BLEU, METEOR, CIDEr, ROUGE, SPICE, WMD. It observes different models used for image and video captioning but they mostly don’t provide the accurate details. In future, we can use the fusion of image, audio and video

combo to provide move accurate captions and which can be used for the people with impairments.

N. Xu et al. [4] uses the multi-level policy and reward Reinforcement learning (RL) framework in the RNN image captioning models. It uses the proposed model on datasets of Flickr, MS COCO. This model evaluates each and every step for better solutions. It also uses the guidance term as bridge which can combine the policy and reward function. The object fails to extract some objects from the images due to lack of clarity in the image. In future, it can solve problem by adding more visualisation models. The evaluation metrics shows the better performance of the model by BLEU 0.805, METEOR 0.271, CIDEr 1.141, ROUGE 0.570.

M. Yang et al. [5] uses the Multitask learning algorithm for cross domain image captioning (MLADIC) which has ability to do two tasks. One is Image captioning and other is text to image conversion and finding the correlation between them for improving the image captioning system. It uses the CNN-LSTM as encoder-decoder model for text generation from images and uses the C-GAN (conditional generative adversarial network) for text to image generation. This model is applied on the MS COCO as main domain, Flickr30k and Oxford-102 are testing domains and performed the best on cross domain image captioning than other existing models.

L. Wu et al. [6] uses the Graph Convolutional networks which helps to capture the background details along with the existing main features. This helps to create a meaningful sentence more accurately. This network consists of both grid graph and region graph which helps to increase the visual context. By conducting the multiple experiments on MSCOCO dataset, this model got the most promising results.

Table 1: Existing Techniques Analysis

S.No	Author	Algorithm Used	Demerits/Future Work	BLEU Scores
1	Leo Ramos, Edmundo Casas.	ConvNeXt, LSTM, visual attention module	Including hyper parameters for enhanced results.	0.75
2	Haya Mesfer Alshahrani, Munya A. Arasi.	Sparrow-Search Algorithm	In future, the computation complexity can be examined.	0.78
3	Ning Xu, Hanwang.	Visual semantic functions, RNN.	More visualized modelling techniques can be added.	0.80
4	Min Yang, Wei Zhao.	Multitask learning algorithm.	The dataset size must be increased and images and text data can be collected form the online sources like wekipedia and printest.	0.9
5	Suya Zhang, Yana Zhang.	Attention Model, Pointer Network.	Accuracy will be increased in the future.	0.91

3. METHODS

Generally, the captions can be easily generated from the images using Deep Neural Networks. The images are passing through the different layers of the network. The image features are extracted easily with these networks. The more layers, the more the features are extracted. Caption features are also extracted with the help of Deep Neural Networks during training. The image features are extracted using the Convolutional Neural Networks (CNN). The caption features are extracted using Long-Short Term Memory (LSTM). The model is the Encoder-Decoder model. The encoder has two parts, one for images and the other for captions. The LSTM is used in the Decoder. The below figure 3 shows the Proposed model.

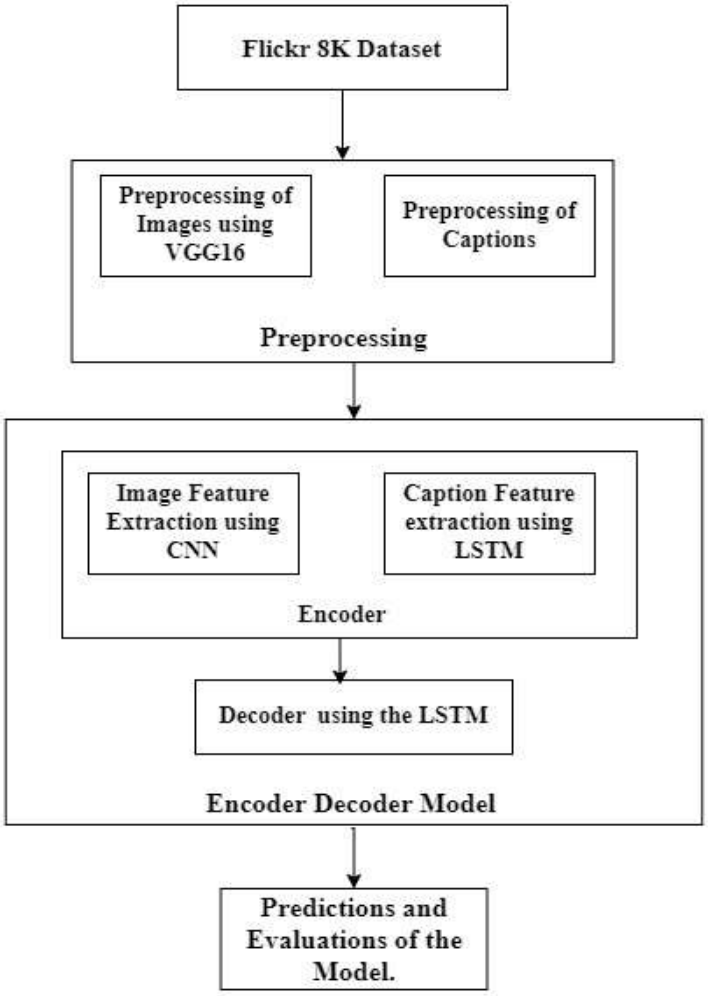

Figure 3: Proposed Model

3.1. Dataset

The Flickr8K dataset used in this model is taken from the Kaggle platform. The dataset consists of two fields one is images and other is captions. Each image consists of five captions related to that image. There are total of 8000 images and it is collected from different Flickr groups. The images are mostly for describing variety scenes and situations.

3.2. Preprocessing of Images using VGG16

Image Preprocessing using VGG16 typically involves resizing the images to match the input size expected by the VGG16 model and applying preprocessing specific to the model, such as mean subtraction. For the Preprocessing of images, the VGG16 does not need the last layer. So, the last layer is not used. Preprocessing images using VGG16 involves resizing them to the model's input dimensions, normalizing the pixel values, and preparing them for input into the model. This ensures that the images are in the correct format and range expected by the VGG16 model, allowing it to make accurate predictions.

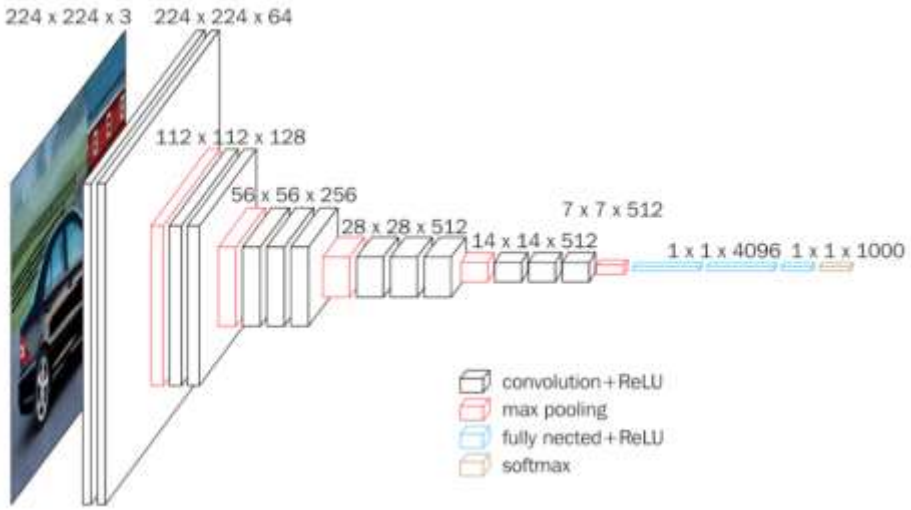

Figure 4: VGG16 Architecture

The above figure 4, shows the VGG16 architecture. It consists of pooling layers, fully connected layers, and convolutional layers. It has a total of thirteen convolutional layers. It has five max pooling layers which are used for dimensional reduction. Also, it has fully connected layers.

Steps in the preprocessing of images:

Step 1. Loading of VGG16 model.

Step 2. Restructure the model.

Step 3. Load the image from file.

Step 4. Convert image pixels to Numpy array.

Step 5. Reshape data from Model.

Step 6. Preprocess image for VGG.

3.3. Preprocessing of Captions

It is necessary to clean the captions as it directly effects the results of the model. The model must be trained on the correct data without any noisy data. The preprocessing of captions is done after creating mapping of images to captions.

Steps in preprocessing of Captions

Step 1. Convert to lowercase

Step 2. Delete digits, special chars, etc.

Step 3. Delete additional spaces

Step 4. Add start and end tags to the caption

3.4. Encoder Decoder Model

Encoder model consists of image feature layers using Convolutional Neural Networks (CNN) and sequence features layers using Long Short-Term Memory (LSTM). Here image features are extracted using the CNN model. The sequence features are extracted using the LSTM. In CNN, the Input layer, Dropout layer, Dense layer are used for feature extraction for images. In LSTM, The Input layer, Embedding-layer, Dropout layer and LSTM layers are used for extracting features of sequences. The decoder model consists of LSTM and dense layer.The below figure 4 will tell us about the layers in the Encoder-Decoder model. For fitting of the model, 20 Epoches are used with the Batch_size of 20.

Steps Involved in Encoder Decoder Architecture

Step 1. The Preprocessed images are sent to the CNN Encoder.

Step 2. The CNN will extract the features and sent to the LSTM.

Step 3. The Feature vector is given input to LSTM and start producing words until end keyword arises or the model cannot able to identify the word.

In the below figure 5, the image can show how image is sent in the encoder decoder model. CNN is used as Encoder and LSTM is used as Decoder.

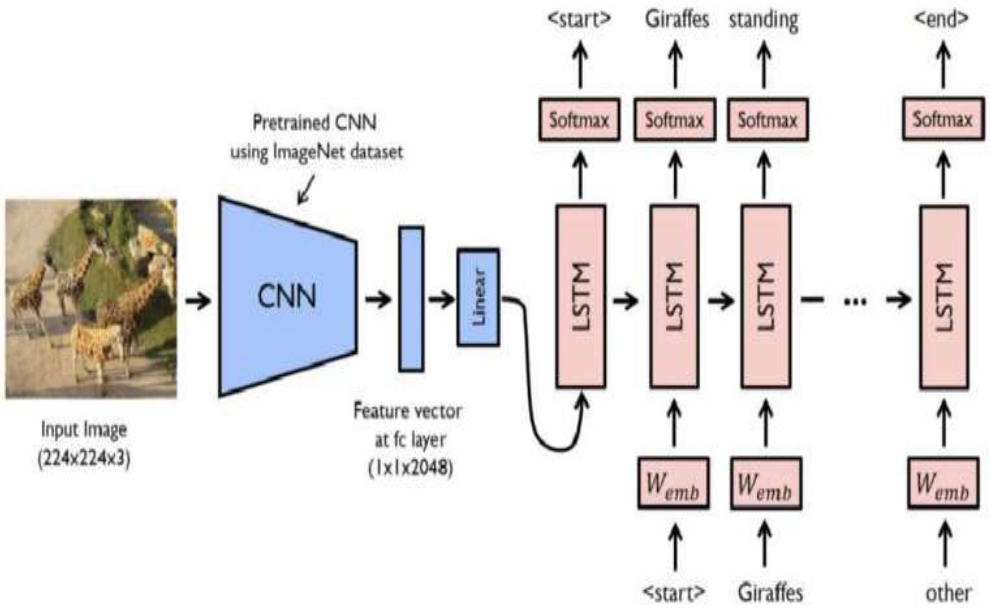

Fig 5: Encoder-Decoder Model Layer Description

In the below figure 6, the Encoder Decoder model is clearly explained. Here the image is sent to the CNN model in which a person is throwing a flying disc in a garden. So, the model will generate the sentence by producing one word at a time. And the produced word along with vector that contains context is send as input to next layer of LSTM which helps to produce next word.

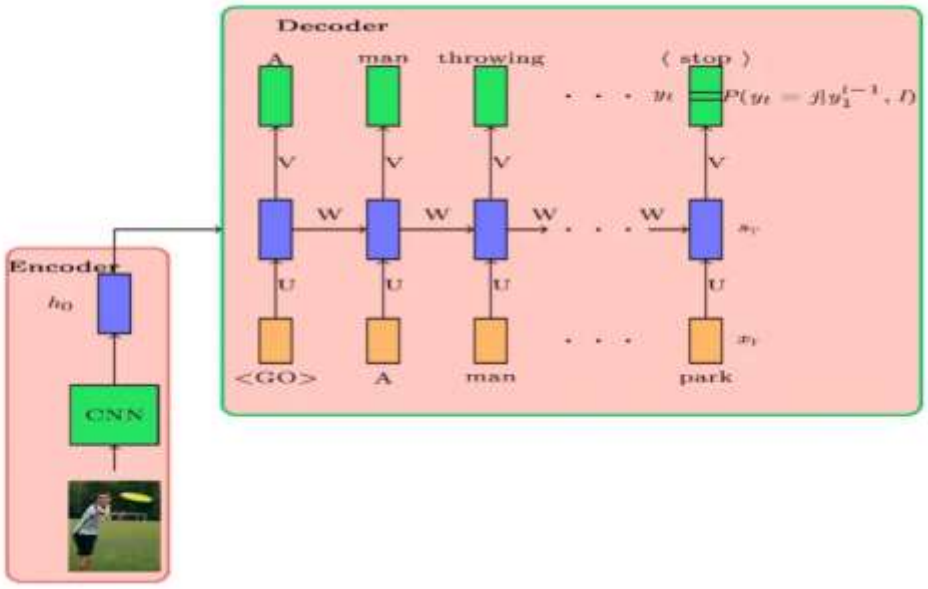

Fig 6: CNN LSTM Architecture

3.5. Evaluation

Bilingual Evaluation Understudy (BLEU) is an evaluation metric used in this model. The value of the score always lies between the 0 and 1. This metric is easy to compute and understand. This uses the unigram and bigram precision scores. This score tells the difference between the real caption and the generated caption. The machine generated text quality is checked by calculation of this score.

4. RESULTS & DISCUSSION

Below figure 7, represents the model summary of the VGG16 in which last fully connected layers are not used because VGG16 is used for the preprocessing of images.

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 224, 224, 3)	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1,792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36,928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73,856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147,584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295,168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590,080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590,080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1,180,160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2,359,808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2,359,808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2,359,808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
fc1 (Dense)	(None, 4096)	102,764,544
fc2 (Dense)	(None, 4096)	16,781,312

Total params: 134,260,544 (512.16 MB)
Trainable params: 134,260,544 (512.16 MB)
Non-trainable params: 0 (0.00 B)
None

Figure 7: VGG16 Model Summary

The preprocessing of text is done by converting the whole text into lower case and deleting the unwanted data from the text. And also deleting the additional spaces along with adding the starting and ending tag for the captions. The below figure 8 can help to show the difference between the text before preprocessing in figure 8a and after preprocessing in figure 8b.

```
['A child in a pink dress is climbing up a set of stairs in an entry way .',
 'A girl going into a wooden building .',
 'A little girl climbing into a wooden playhouse .',
 'A little girl climbing the stairs to her playhouse .',
 'A little girl in a pink dress going into a wooden cabin .']
```

Figure 8a: Text before preprocessing

```
['startseq child in pink dress is climbing up set of stairs in an entry way endseq',
 'startseq girl going into wooden building endseq',
 'startseq little girl climbing into wooden playhouse endseq',
 'startseq little girl climbing the stairs to her playhouse endseq',
 'startseq little girl in pink dress going into wooden cabin endseq']
```

Figure 8b: Text after preprocessing

Figure 8: Preprocessing of text in captions with respect to each image

In the below figure 9, it explains the Encoder-Decoder model layers used in the proposed model. This model consists of different layers. The Input layer will take all the input images and it is the first layer of this Neural Network. The embedded layer is hidden and helps in dimensionality reduction. The dense layer has full connections with the previous layer. It is connected with every neuron of previous layer. The dropout layer transfers only important features to the next layer. That layer makes the unwanted neurons to zero.

Model Formulas:

- Encoder

$$S_0 = CNN(x_i)$$

- Decoder

$$S_t = RNN(S_{t-1}, e(y_{t-1}^{\wedge}))$$

$$P(y_t | y_1^{t-1}, I) = softmax(V_{S_t} + b)$$

E(y_hat(t-1)) is the one hot encoded form of the output at (t-1) th time step.

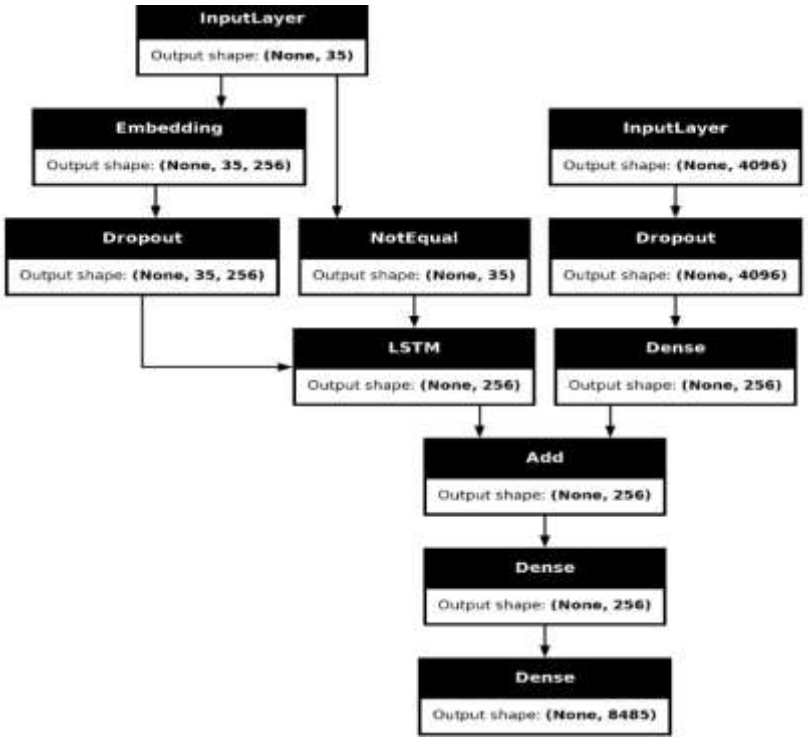

Figure 9: Summary of Encoder Decoder Model

The generation of captions means predictions of sentences for the images. These captions are generated by the Encoder Decoder model after training is completed to check the evaluation of the model that is created. There are some to be followed to predict the sentence.

Steps for generating captions for images

Step 1. Give the image as input without the captions.

Step 2. Encode the input sequence.

Step 3. Pad the sequence.

Step 4. Get an index with high probability.

Step 5. Convert index to word.

Step 6. Stop if word not found.

Step 7. For generating next word, the appending of words in input must be done. And finally, terminate by encountering the end tag.

Below image figures 10, 11, 12 can show us the differentiation of actual captions and the predicted captions. There are five actual captions and one predicted caption.

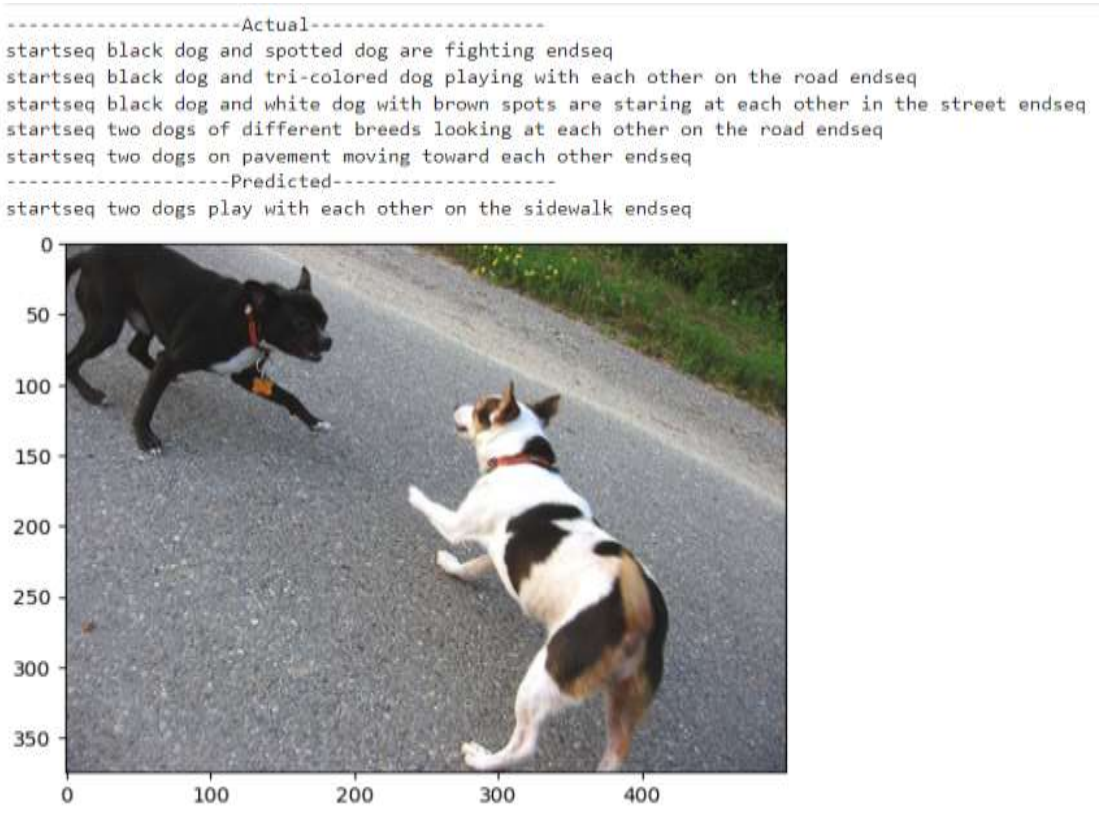

Figure 10: Caption predicted for 1001773457_577c3a7d70.jpg

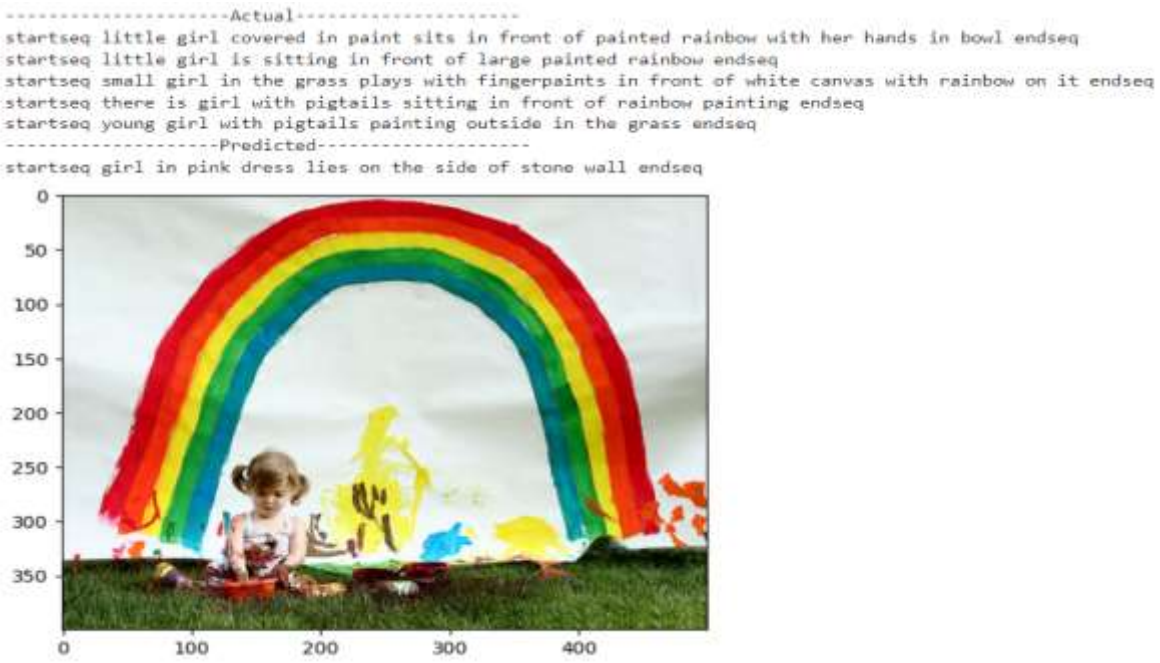

Figure 11: Caption predicted for 1002674143_1b742ab4b8.jpg

Figure 12: Caption predicted for 101669240_b2d3e7f17b.jpg

BLEU has two classifications. Unigram means a single words. Bigrams means the two words together. BLEU - 1 was used to calculate the precision score of single words called unigrams. BLEU - 2 is the average score that is calculated for unigram and bigram precision score. The precision score checks how correctly the model is predicting.

For calculation of BLEU score, first convert the predicted caption and references to unigram/bigrams.

$$BLEU(N) = Brevity\ Penality\ \times Geometric\ Average\ Precision\ Scores\ (N)$$

Where the N value is mostly taken as 4. And formulas of Brevity Penality and Geometric Average Precision are as follows

$$Brevity\ Penality = \begin{cases} 1, if\ c > r \\ e^{(1-r/c)}, if\ c \leq r \end{cases}$$

Where c is predicted length and r is target length.

$$Geometric\ Average\ Precision\ (N) = \exp(\sum_{n=1}^{N} \omega_n \log p_n)$$

Where w means uniform weights and p means individual precision values.

5. CONCLUSION

The Proposed Model for Image Captioning can generate the captions automatically without any disturbances. It can also able to produce long sequences using the LSTM-CNN model. The preprocessing of images can be accomplished using the VGG16 and removes the noise. The preprocessing of captions helps to reduce the ambiguity in text generation. The Encoder-Decoder model consists of CNN and LSTM models. Those models are used to extract the features of images and text. LSTM is a Recurrent neural network (RNN) technique that makes the model for the prediction of the next words in the sentences. The BELU metric is used for the evaluation of the model. This proposed model outperforms other existing models and achieves better results in generating captions. In the future, the training of the model can be improved and increase the evaluation score. Also, this model can be trained on different domains to increase the range of caption generation.

REFERENCES

[1] L. Ramos, E. Casas, C. Romero, F. Rivas-Echeverría and M. E. Morocho-Cayamcela, "A Study of ConvNeXt Architectures for Enhanced Image Captioning," in IEEE Access, vol. 12, pp. 13711-13728, 2024, doi: 10.1109/ACCESS.2024.3356551.

[2] M. A. Arasi, H. M. Alshahrani, N. Alruwais, A. Motwakel, N. A. Ahmed and A. Mohamed, "Automated Image Captioning Using Sparrow Search Algorithm With Improved Deep Learning Model," in IEEE Access, vol. 11, pp. 104633-104642, 2023, doi: 10.1109/ACCESS.2023.3317276.

[3] S. Amirian, K. Rasheed, T. R. Taha and H. R. Arabnia, "Automatic Image and Video Caption Generation With Deep Learning: A Concise Review and Algorithmic Overlap," in IEEE Access, vol. 8, pp. 218386-218400, 2020, doi: 10.1109/ACCESS.2020.3042484.

[4] N. Xu et al., "Multi-Level Policy and Reward-Based Deep Reinforcement Learning Framework for Image Captioning," in IEEE Transactions on Multimedia, vol. 22, no. 5, pp. 1372-1383, May 2020, doi: 10.1109/TMM.2019.2941820.

[5] M. Yang et al., "Multitask Learning for Cross-Domain Image Captioning," in IEEE Transactions on Multimedia, vol. 21, no. 4, pp. 1047-1061, April 2019, doi: 10.1109/TMM.2018.2869276.

[6] L. Wu, M. Xu, L. Sang, T. Yao and T. Mei, "Noise Augmented Double-Stream Graph Convolutional Networks for Image Captioning," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 8, pp. 3118-3127, Aug. 2021, doi: 10.1109/TCSVT.2020.3036860.

[7] P. Mahalakshmi and N. S. Fatima, "Summarization of Text and Image Captioning in Information Retrieval Using Deep Learning Techniques," in IEEE Access, vol. 10, pp. 18289-18297, 2022, doi: 10.1109/ACCESS.2022.3150414.

[8] M. Z. Hossain, F. Sohel, M. F. Shiratuddin, H. Laga and M. Bennamoun, "Text to Image Synthesis for Improved Image Captioning," in IEEE Access, vol. 9, pp. 64918-64928, 2021, doi: 10.1109/ACCESS.2021.3075579.

[9] C. Chen et al., "Towards Better Caption Supervision for Object Detection," in IEEE Transactions on Visualization and Computer Graphics, vol. 28, no. 4, pp. 1941-1954, 1 April 2022, doi: 10.1109/TVCG.2021.3138933.

[10] Y. Xu, W. Yu, P. Ghamisi, M. Kopp and S. Hochreiter, "Txt2Img-MHN: Remote Sensing Image Generation From Text Using Modern Hopfield Networks," in IEEE Transactions on Image Processing, vol. 32, pp. 5737-5750, 2023, doi: 10.1109/TIP.2023.3323799.

[11] X. Li, A. Yuan and X. Lu, "Vision-to-Language Tasks Based on Attributes and Attention Mechanism," in IEEE Transactions on Cybernetics, vol. 51, no. 2, pp. 913-926, Feb. 2021, doi: 10.1109/TCYB.2019.2914351.

[12] S. Zhang, Y. Zhang, Z. Chen and Z. Li, "VSAM-Based Visual Keyword Generation for Image Caption," in IEEE Access, vol. 9, pp. 27638-27649, 2021, doi: 10.1109/ACCESS.2021.3058425.