Image Caption Generator with Deep Learning

B. Nandini

ABSTRACT

The process of creating descriptions for the events depicted in an image is known as image captioning. Deep Learning Models can be used to accomplish this image captioning. It is an extremely difficult issue to automatically generate a caption or explanation for an image using any natural language sentence. It takes techniques from computer vision to comprehend the image's content and a language model from natural language processing to translate the comprehension of the image into words in the correct sequence.

Deep learning and natural language processing have advanced to the point where creating captions for the provided photos is now simple. We use a Convolutional Neural Network (CNN) that has been trained beforehand to extract high-level features, such as objects, forms, and textures, from photos. A Long Short-Term Memory (LSTM) network, a kind of Recurrent Neural Network (RNN) that can handle sequential input like sentences, is then fed these features.

Keywords: Long Short-Term Memory Network (LSTM), Convolutional Neural Networks, and Deep Learning

INTRODUCTION

Image captioning used to be a difficult task, and the captions that are created for a specific image are often not very useful. Many tasks that were tough and difficult to perform using machine learning became simple to apply with the help of Deep Learning and Neural Networks thanks to the evolution of these technologies. These approaches include Natural Language Processing and Neural Networks for Deep Learning. These are highly helpful for many different applications using artificial intelligence, such as picture recognition, image classification, and image captioning. The process of creating descriptions for the events depicted in an input image is known as image captioning. Many models, including object detection models, visual attention-based models, and deep learning models, have been proposed for image captioning. There are several deep learning models in deep learning as well, including the Inception model, the VGG model, the ResNet-LSTM model, and the conventional CNN RNN model. This paper will provide an explanation of the captioning model we used for the photographs.as in the CNN-LSTM model.

METHODS

Dataset Information

The FLICKR 8K data collection is being used in this project to train the model. The FLICKR 8K data collection is effective in training the Deep Learning Model for Image Caption Generating. There are 8000 photos in the FLICKR 8K data set, of which 6000 can be used to train the deep learning model, 1000 for development, and 1000 for testing. Each image in the Flickr Text data collection has five captions that explain the activities taken in that particular photograph. Predicting the captions for the input image is the project's goal. Eight thousand photos make up the dataset, and each image has five captions. For input, the features are taken from the written captions and the image. The following word in the caption will be predicted by concatenating the traits. For images, CNN is utilized, and for text, LSTM. A statistic called the BLEU Score is employed to assess the trained model's performance.

Image preprocessing

To make sure the photos can be fed into the VGG19 model, we must preprocess them after loading the datasets. All of the photos must be resized to 224x224x3, as convolutional layers such as those in VGG19 require constant image dimensions. We'll also convert the pictures to the RGB color space.

Text preprocessing

We must preprocess the captions after loading them using the FLICKR text data set to ensure that there are no ambiguities or difficulties when the deep learning model is being trained or when creating vocabulary from the captions.After determining whether any numbers are present in the captions, we must eliminate them. Next, we must eliminate any white space and any missing captions from the provided data set.To remove uncertainty when expanding the vocabulary and training the model, we must convert all uppercase letters in the captions to lowercase.

Since this model creates captions one word at a time, previously created words are used as inputs along with the features of the images. These words are attached to the beginning and end of each caption to inform the neural network where to start and stop the captions while the model is being trained and tested.

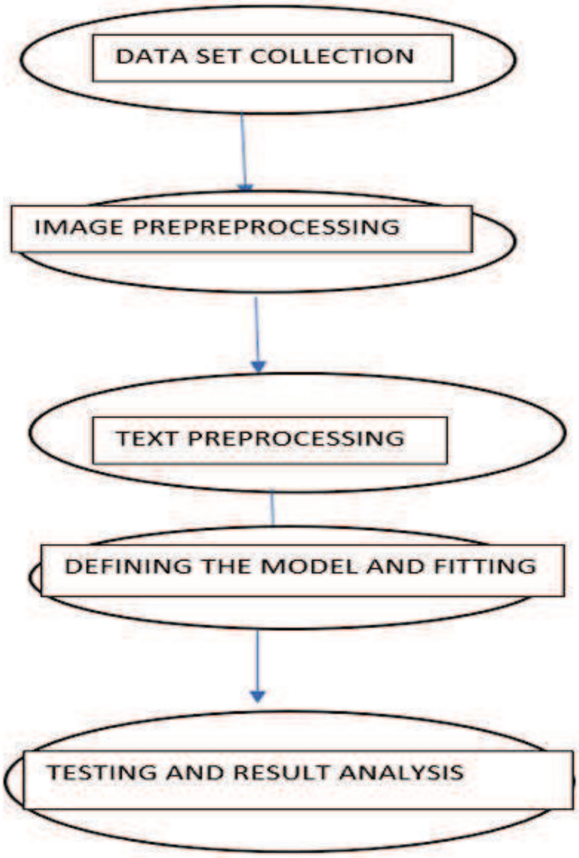


Fig 1: Model Implementation Flow Chart

THE ROLE OF CNNS IN IMAGE CAPTIONING

1. Image Preprocessing:
   - Preprocessing is done on images before they are fed into CNN. To establish consistency for the model, this may entail scaling, normalizing, and color space conversion.
2. Feature Extraction:
   - There are numerous convolutional layers and pooling layers in the CNN architecture. Convolutional layers use filters to scan the picture and identify local patterns, forms, and edges. The CNN gradually extracts more complex features from the image through pooling layers, which downsample the data while maintaining key features. These features range from simple edges to more complicated objects and their spatial relationships.
3. Feature Map as Output:
   - The last convolutional layer, which processes the image, produces a feature map, which is a condensed representation of the image that highlights its most important visual components. The objects, situations, and their spatial arrangements make up the "what" of the image, and these are fundamentally encoded in the feature map.

The Power of LSTMs for Caption Generation

1. Understanding Sequences:

A particular kind of Recurrent Neural Network (RNN) called Long Short-Term Memory (LSTM) is made to handle sequential data, such as sentences. Because of this, they are perfect for picture captioning, where the objective is to come up with a list of words that best describe the content of the image.

2. Processing the Feature Map:

The LSTM network receives this feature map, which provides a condensed representation of the "what" in the image. The CNN extracts a feature map summarizing the visual elements of the image.

3. Decoding Features into Words:

The LSTM can retain information longer than conventional RNNs because of its special architecture, which includes memory cells. This is crucial for captioning images because the model has to take the picture as a whole and its context into account when producing the caption.

The current input from the feature map and the previous hidden state, which contains details about previously created words, are fed into the LSTM at each step.The LSTM determines which word is most likely to appear next in the caption based on this information.

4. Word-by-Word Generation:

After adding the anticipated word to the caption, the LSTM advances to the following stage.Iteratively, the LSTM predicts the next word and builds the caption word by word using the feature map and the expanding caption.

CNN-LSTM ARCHITECTURE

This problem was found when training conventional RNNs because, as neural networks get deeper, little to no training may take place if the gradients are very small or zero, which leads to poor prediction performance. LSTM networks are ideally suited for classifying, processing, and forecasting based on time series data because there could be lags of unknown length between important events in the series. LSTM outperforms the standard RNN in terms of efficiency and superiority since it gets around the RNN's short-term memory limitations.

The LSTM has the ability to process inputs while ignoring irrelevant data and processing useful information. Even after they have been defined in the abstract, define acronyms and abbreviations whenever they are used for the first time in the text. There is no need to define acronyms like IEEE and SI. If they are not avoidable, do not use abbreviations in the title or headers. The CNN-LSTM architecture uses CNN layers for feature extraction from input data and LSTMs to help in sequence prediction. This method is particularly designed for sequence prediction issues including spatial inputs, such images or videos. They are widely used in a variety of tasks, such as activity recognition and the description of images and videos.Fig. 3 depicts the CNN-LSTM Model's overall architecture:

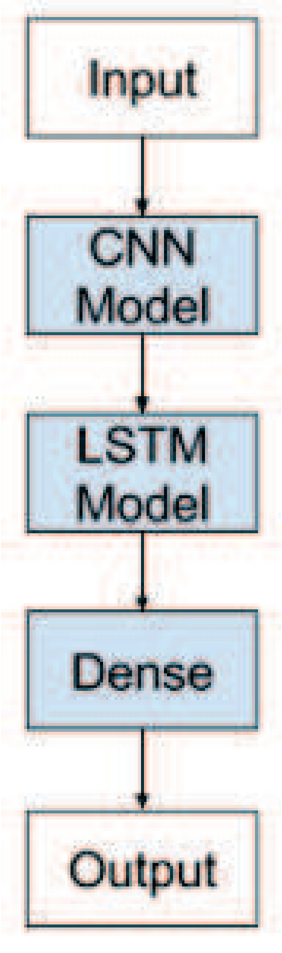


Fig 2 : General Architecture of CNN- LSTM Model

IMPORT MODULES

os - used with system commands to manage files.

pickle - utilized to store extracted numpy features

numpy – a Python module that works with arrays to execute a variety of mathematical calculations.

tqdm - Iterator decorator for progress bars. features an iterator with a preset range that prints to STDERR.

VGG16, preprocess_input - modules that are imported to extract features from the picture data

load_img, img_to_array – used to load the picture and turn it into a numpy array.

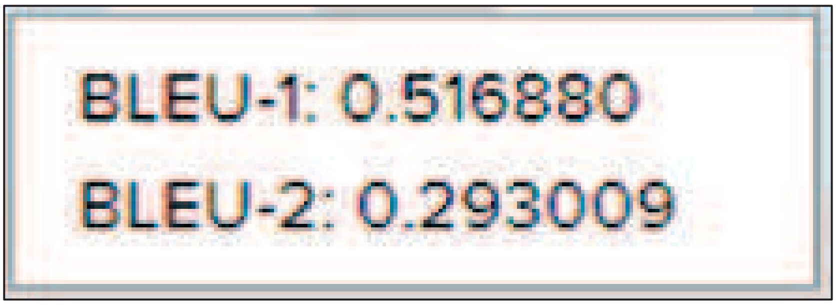
Tokenizer - utilized to load the text and turn it into a token

pad_sequences - utilized to distribute words evenly between phrases, filling in the empty spaces with zeros.

plot_model - used to display the model's architecture as a series of photos

VALIDATE THE DATA USING BLEU SCORE

Dual Assessment The descriptions produced by translation and other Natural Language Processing (NLP) applications are assessed using the Understudy or BLEU score. The BLEU Score is used to compare the predicted text with a reference text in a list of tokens. The reference text contains every word that is inserted from the real captions data. Increase the number of epochs in accordance with the desired score, with a BLEU Score of larger than 0.4 being regarded as good. Figure 11: The input image's BLEU score.



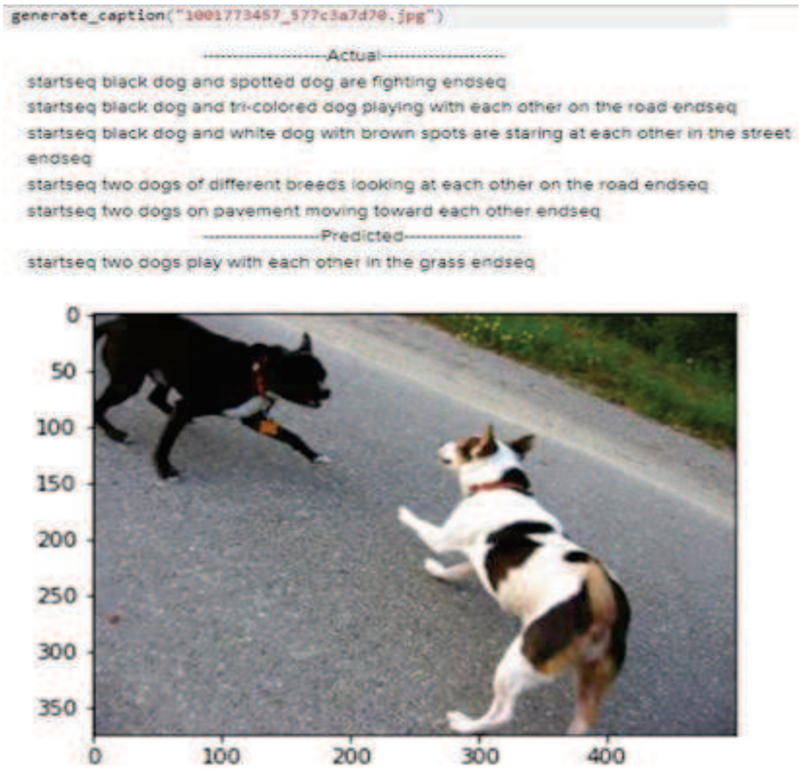
```
BLEU-1: 0.516880
BLEU-2: 0.293009
```

Fig 3 : BLEU Score for input image

VISUALIZE THE RESULTS

The model's ability to accurately describe the content of different images is demonstrated by the visible results for the image caption generator. This highlights the efficacy of combining CNNs for feature extraction with LSTMs for sequential language generation.

Following model definition and fitting. It is noted that in the early training epochs, the accuracy is quite poor and the generated captions have little to do with the test images. We have found that the generated captions are somewhat linked to the provided test photos when the model is trained for at least 20 epochs.



```
generate_caption("1001773457_577c3a7d70.jpg")
```

```
                    --------------------Actual--------------------
  startseq black dog and spotted dog are fighting endseq
  startseq black dog and tri-colored dog playing with each other on the road endseq
  startseq black dog and white dog with brown spots are staring at each other in the street
  endseq
  startseq two dogs of different breeds looking at each other on the road endseq
  startseq two dogs on pavement moving toward each other endseq
                    --------------------Predicted--------------------
  startseq two dogs play with each other in the grass endseq
```

Fig 4 : Captions generated of the given Image

CONCLUSION

This paper presents the method we learned and developed for creating an Image Caption Generator that can provide the user with descriptions or captions based on an image. The Language Based Model converts the objects and image features into English phrases. The Image Based Model extracts picture features. The language-based model uses LSTM, while the image-based model uses CNN. Increasing the number of epochs in the model's training can produce better, more accurate results.

It can take a long time and require a lot of system resources to process big data. We can increase the model's layer count if we wish to handle big datasets like flickr32k. We can extract picture features using the CNN model in addition to the VGG19 model. Data gathering, pre-processing, model training, and prediction comprise the workflow. Through automatically generated captions or descriptions, picture caption generators hope to enhance social networking platforms, image indexing, and accessibility for individuals with visual impairments.

REFERENCES

1. https://research.google.com/colaboratory/faq.html
2. https://towardsdatascience.com/introduction-to-kaggle-kernels-2ad754ebf77
3. https://www.geeksforgeeks.org/introduction-to-pandas-in-python/
4. https://www.geeksforgeeks.org/numpy-in-python-set-1-introduction/
5. https://www.geeksforgeeks.org/python-introduction-matplotlib/
6. https://www.javatpoint.com/keras
7. "Image doi: Captioning Using Deep Convolutional Neural Networks (CNNs)" by G. Geetha, T. Kirthigadevi, G. Godwin Ponsam, T. Kartik, and M. Safa 2015 Journal of Physics:Conference Series, Volume 1712, International Conference On Computational Physics in Emerging Technologies (ICCPET) 2020 August 2020, Manglore, India, published under license by IOP Publishing Ltd.