IMAGE CAPTION GENERATOR

Divya Reddy
dept. Computer Science
University of Ottawa
dredd037@uottawa.ca
300290332

Syeda Zainab
dept. Computer Science
University of Ottawa
szain039@uottawa.ca
300303434

Ishveen Manjeet Singh Sahi
dept. Computer Science
University of Ottawa
isahi065@uottawa.ca
300303960

***Abstract*—In today's world, it is very important to have a captions generator as it allows the creation of image captions that helps to label and describe an image. Image captioning is the process of creating a description for an image and knowing what the image states or depicts. In this process, the image is recorded and translated into a plain language that is human-understandable. This process is a time-consuming task that requires combining image identification and computer vision. In our proposed project, we use CNN and LSTM model that uses computer vision and machine translation to describe images and generate captions.**

The goal of this project is to find important objects in an image, recognize the relationships in those objects and predict the sentence sequence using LSTM.

Keywords—CNN, LSTM, Captions generator, Computer vision

I. INTRODUCTION

IMAGE CAPTIONING

Image captioning is a process that creates textual descriptions for images, these descriptions are called captions that are generated using computer vision and language processing. Image captioning can be considered an end-to-end (sequence-to-sequence) problem as it converts images, it processes a sequence of pixels to a sequence of words.

To develop image captions the sequence-to-sequence model is used, these models work by generating a representation by RNN based on an input sequence, and then this input is fed into a second RNN as an output representation to generate another sequence. It's important to find and identify objects in images. It must also comprehend the type of scene or location, object properties, and interactions between objects. It takes knowledge of the language's syntactic and semantic structures to produce well-formed sentences.

OBJECTIVE

We see a lot of images every day from various sources, including the internet, social media, diagrams in documents, news articles, and advertisements. The images in these sources are for the viewers to interpret. The majority of images in these sources lack captions and descriptions, but a human being can still understand them in large part without them. But in order for humans to use automatic image captions from it, machines must be able to understand a variety of image captions.

For a number of reasons, image captioning is very significant as it acts as a source for people and machines to know about a particular image. Internet image searches and indexing can be accelerated and made more descriptive with captions for each image.

Creating captions for images is an important task that is relevant to both ComputerVision and Natural Language Processing. A machine mimicking the human ability to provide descriptions for images is a remarkable step forward in the field of Deep Learning. The main challenge of this task is to capture how objects interact in the image and express them in natural language.

Computer systems have traditionally used predefined templates to generate text descriptions for images. However, this approach needs to provide more variety to generate lexically rich text descriptions.

The main objectives of our paper are

- By using the convolution neural network technique to extract features from the images given.

- To train a model based on the given dataset such that it can detect various elements in that particular image such as objects, scenes, persons, etc.

- To train a model to generate captions and descriptions for the given image at a very level. Such as character, word, or sentence level.

MOTIVATION

The significance of image caption generators is frequently understood by their applications. Among the applications where the solution to this problem might be useful is

- Aid to the blind

  We can use image captions to describe the environment to a blind person or a person with a low- vision person, people who rely on sound and text to describe a particular scene or space.

- Google Image Search

  Automatic image captioning for images can help google image search more as it would result in better search results and will also be based on the captions provided as well.

- Self-driving cars

  Nowadays automatic driving is one of the biggest challenges and image captioning will be an asset if the environment around the car can be captioned and described correctly as it will benefit the self-driving system.

- CCTV cameras

  There is a need for CCTV cameras all over the world as there is a relevant use case, so if image caption generators are used to generate captions and descriptions then alarms can be raised faster and quicker if there is any malicious activity going on anywhere.

II. LITERATURE REVIEW

Using the ideas of a Long Short-Term Memory and Convolutional Neural Network a model is created to detect image captions, While the LSTM extracts the word vectors, CNN extracts the image vectors.[1]

In this paper, they used three image captioning techniques. CNN-RNN-based and CNN-CNN-based models were primarily used in this paper and they provided sample works and evaluation metrics and discussed in detail the disadvantages of the three systems. [2]

In this paper, they used two approaches to Image Captioning: bottom-up and top-down. Top-down approaches were considered in CNN-LSTM architecture, which was modeled after the NIC architecture, by using a deep convolutional neural network to generate a victory representation of an image and then using Long Short-Term Memory(LSTM) captions generated. Overfitting is reduced by using hyperparameter tuning with dropout and the number of LSTM layers. A deep convolutional neural network is used to generate a victory representation of an image, and the Long-Short-Term Memory (LSTM) network is used to generate captions.[3]

In this paper, the proposed model can generate more unique and novel captions than the baseline in all three datasets used in this experiment. When compared to methods that only use CNN as an encoder, the model performed better. When image captions are decoded in a phrase-based hierarchical manner, all sub-metrics improve. The average length of captions generated by the phi LSTM model is shorter when compared to baseline models. Baseline models correctly predicted more words in the Flickr 8k dataset than the model proposed in this paper.[4]

The aforementioned research paves the way for improving models to create image captioning systems. The most practical and successful method for captioning an image through a dataset is to use CNN and RNN.

By using the Flickr8k datasets for training and obtaining trained dataset weights that can be used for image captioning, our contribution to the existing models is made possible. Image caption generation and image recognition in self-driving cars are two examples of useful features for people who are blind or visually impaired.

III. PROPOSED METHODOLOGY

The objective is to develop a system that can take an image in the form of a dimensional array, characterize it, and then output syntactically and grammatically sound statements.

1. CONVOLUTIONAL NEURAL NETWORK (CNN)

Convolutional Neural Networks (CNN) is a subset of Deep Neural Networks, a class of artificial neural networks that are effective at gleaning information from visual data.CNN's are heavily deployed in Computer Vision applications and while processing pixel data.

CNN majorly comprises four hidden layers-

1. Convolutional layers:

   The Convolutional layer comprises two input layers:

   - Part or Segment of the input image.
   - Kernel - which is the filter of the input image which is captioned.

2. Pooling layers:

   The Pooling layer is used to down-sample the image data. It decreases the number of pixels of the input image thereby reducing the size of the input image. Pooling can be performed in two ways either by Max Pooling or Min Pooling. The maximum value from the focused region is chosen in the former while the minimum value is chosen in the latter technique.

3. Fully Connected layers:

   The input of the other levels is connected to the output of one layer. For tasks requiring classification, these layers perform incredibly well.

4. Normalization layers:

   The Normalization Layer is to stabilize the neural networks. Normalization is performed on the input images and is used to normalize the intermediate layers.

A Convolutional Neural Network takes an image as input, assigns importance to various objects in the image, and distinguishes between them. Convolutional neural networks can process data in the form of a 2D matrix. Images can be easily represented as a 2D matrix, and CNN is an excellent tool for working with them. The photos are scanned from left to right and top to bottom to extract important elements before combining them to classify.

2. LONG SHORT-TERM MEMORY (LSTM)

Recurrent Neural Networks (RNN) is harnessed to generate sequences of textual data in the form of words or phrases. RNNs are typically the caption-generating component of our model. The extracted image features are usually passed to the RNN to generate textual information about it.

Long short-term memory (LSTM) is a type of RNN (recurrent neural network) that is well suited for solving series prediction problems. By overcoming the limitations of RNNs with short time periods of memory, it has proven to be more powerful than traditional RNNs. While processing inputs, LSTM can perform appropriate statistics and discard non-applicable statistics using an overlook gate. [10]

The LSTM network is capable of remembering lengthy sequences of words and enables forming of textual descriptions of the information extracted from the input image. Deep convolutional neural networks are used to form semantic representations of the image which are then decoded by the LSTM network.

DATASET

The Flickr 8k dataset is a publicly accessible standard for image-to-sentence description. There are 8091 photographs in this collection, each with five captions. These images were sourced from various Flickr groups. Each caption goes into great detail about the objects and events depicted in the photograph.

- Flickr 8k.token contains all the image names with the captions
- Flicker8k_Dataset folder - 8091 images are stored
- Flickr_8k_text folder - text files with captions of images are stored

The model has a small size so it is comparatively easier for it to be trained on lower-end laptops. It is also free and well-labeled.

WORKING MODEL

CNN Neural Networks is used in our paper for image caption generator:

- The image features will be supplied by the LSTM model, which will be in charge of creating captions for the images.

- The CNN model Xception is trained on the Flickr8k dataset.

- The input is in the form of a 2D matrix. It's easy to create images. It has the capacity to manage the uploaded photos.

- Long short-term memory (LSTM) can be translated, rotated, scaled, and shifted in perspective.

The Image Caption Generator comprises the following Deep Learning Neural Networks:

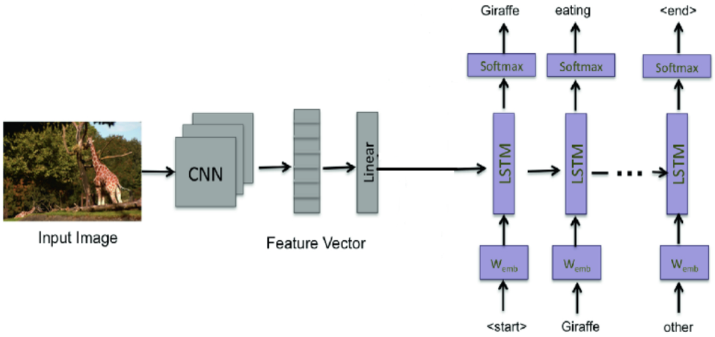


Figure 1: Image Caption Generator model [5]

Image Caption Generator Model(CNN-RNN model) = CNN + LSTM

- CNNs- To extract spatial information in the form of features from the input images.
- RNNs- To generate sequential data of words and phrases.
- LSTM- To generate a description from the extracted information of the image.

Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) models are used in our proposed paper.

Encoder - CNN will act as an encoder to extract features from a given image.

Decoder - LSTM will act as a decoder to generate words that describe the given image.

The LSTM model is incapable of understanding the given directly fed images because LSTM is not developed to handle such inputs; there is a requirement for an RGB image tensor. To predict anything using the LSTM model there is a need to extract some features from the images which can then be fed into LSTM architecture. [11]

It is split majorly into three main sections:

1. Feature Extractor: Using a dense layer, we can reduce the 2048 nodes in the extracted feature from the image to just 256 nodes.

2. Sequence Processor: An embedding layer will handle the text input, and then the LSTM layer will take over.

3. Decoder - By combining the results from the first two layers, we will make the final prediction. The number of nodes in the final layer will equal the size of our vocabulary.

IV. IMPLEMENTATION

The implementation of the image caption generator is shown in Figure 2. Firstly a given data input image is loaded and then the features are extracted from the given image. After this, the caption text file is loaded and the data is preprocessed as well as tokenized. After tokenization, the image mapping to its caption is saved in a separate file.

The given data image is then encoded using the pre-trained classification model and after this, the model is trained and evaluated. Following this, the caption produced is most closely related to the output.

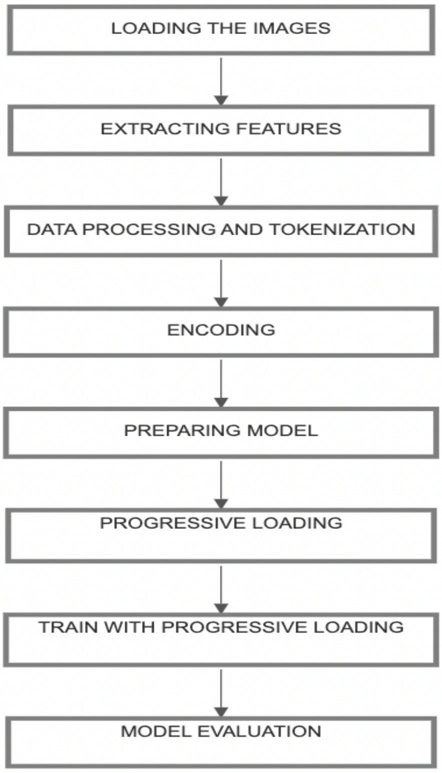


Figure 2: Stages of captioning image

1. **Loading the images:** The image is loaded with the help of Keras. It provides a function that loads the images as a matrix, which is then transformed to a NumPy array by Keras and loads the images in a fixed dimension.

2. **Extracting features:** To classify the images, we use a pre-trained model called Xception, which was previously trained with an imagenet dataset with 1000 different classes. Because the Xception model requires a 299*299*3 image size as input, we must remove the last classification layer and extract the 2048 feature vectors. Xception is faster to train as the file is small ~96MB.

3. **Data preprocessing and tokenization:** Descriptions must be tokenized and easily handled. In this task Converting to lowercase, removing punctuation, removing unnecessary words, and removing numbers are all ways to clean up. All images present in the Flickr Dataset have five captions.

4. **Encoding:** Prior to processing, the words must first be encoded. To prepare the descriptions, Keras is employed. In order to start, we map the image identifiers to the existing descriptions.

   We have 4485 words for the model, 8091 photos, and 28 words (length of description) after this process is complete (vocabulary dataset).

5. **Preparing model:** The sentence is generated word-by-word using a model. The input for the image is the image and the recently predicted word, and the model is referred to as recursively.

   It uses the previously predicted words to generate the new words and uses input and output pairs; the new words are predicted by probability. We have trained the model for 4 epochs which would be good for testing our images.

6. **Progressive loading and training:** With 6000 training images, create the input and output sequences to train the model.

In order to fit the batches into the model, we develop a function. The model is then saved to our model's folder.

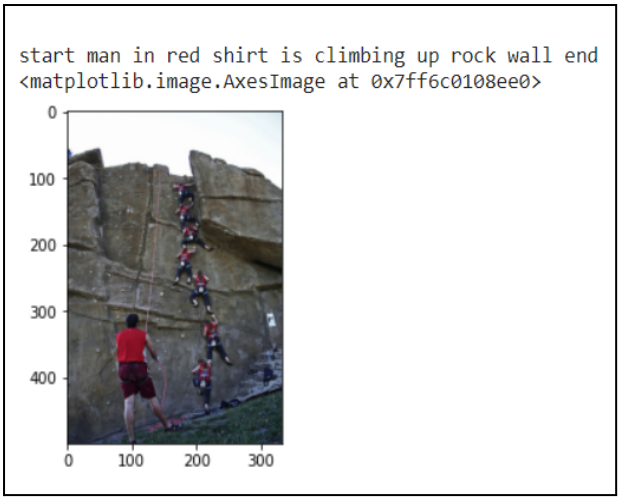
7. **Model evaluation:** After the model has been trained, a separate file is created that will load the model and produce predictions. We use the same tokenizer.p pickle file to extract the words from their index values because the predictions contain the maximum length of index values.

V. RESULTS

Few of the images passed to our model are shown in this section along with the predicted captions made by our trained model. Please refer to Table 1 for further details.

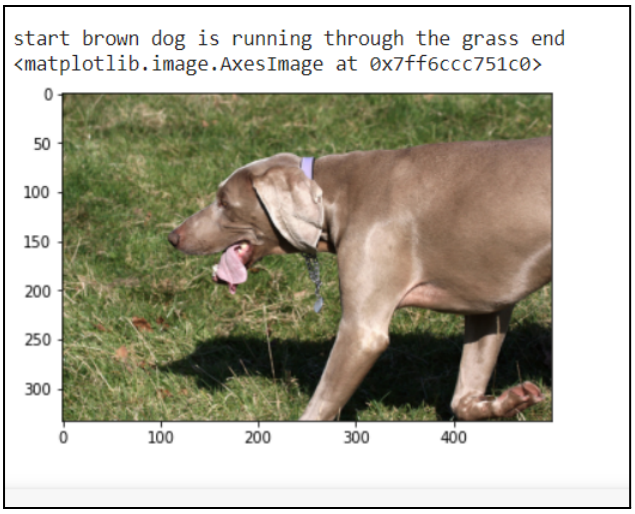
IMAGE	DESCRIPTION
Image 1 - Figure 3	Man in the red shirt is climbing up the rock wall
Image 2 - Figure 4	Brown dog is running through the grass end
Image 3 - Figure 5	brown dog is running through water end
Image 4 - Figure 6	Man in the red shirt is jumping in the water end
Image 5 - Figure 7	Two children are playing in the water end

Table 1: Details of the images used as input to generate captions



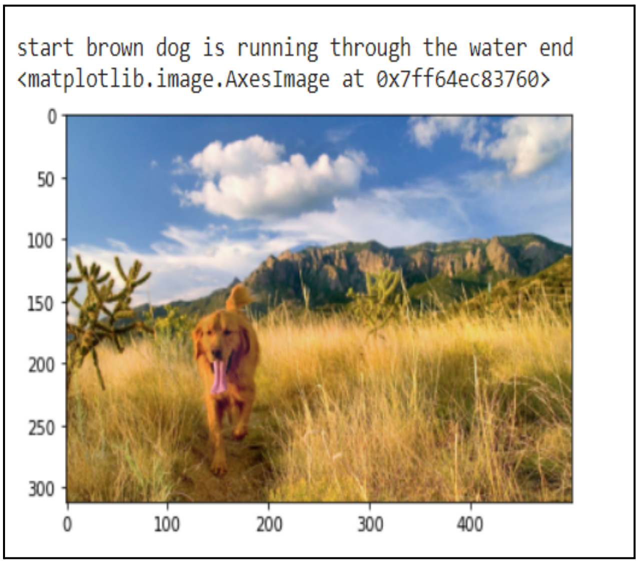
```
start man in red shirt is climbing up rock wall end
<matplotlib.image.AxesImage at 0x7ff6c0108ee0>
```

Figure 3: Man in red shirt image



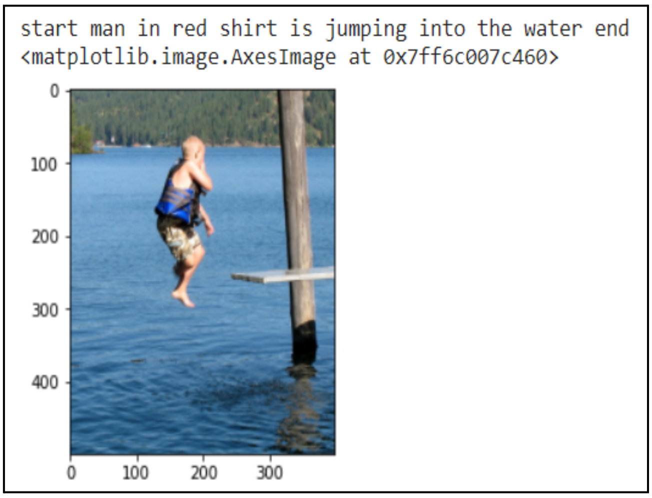
```
start brown dog is running through the grass end
<matplotlib.image.AxesImage at 0x7ff6ccc751c0>
```

Figure 4: Brown dog image



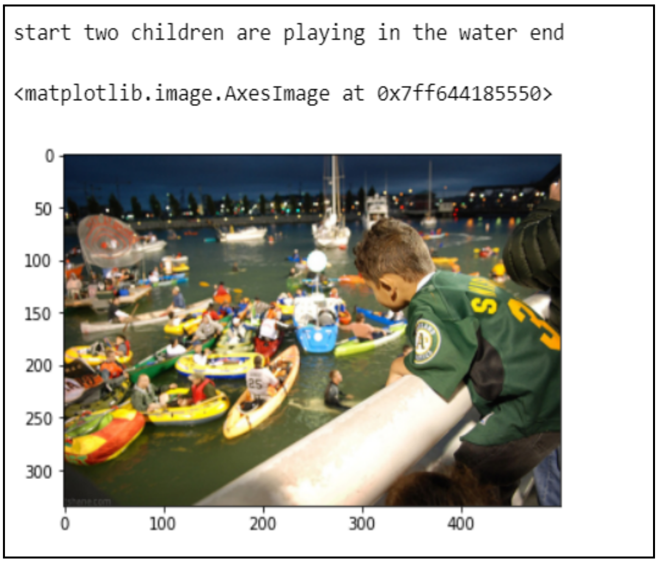
```
start brown dog is running through the water end
<matplotlib.image.AxesImage at 0x7ff64ec83760>
```

Figure 5: Brown dog running image



```
start man in red shirt is jumping into the water end
<matplotlib.image.AxesImage at 0x7ff6c007c460>
```

Figure 6: Man jumping in water image



```
start two children are playing in the water end

<matplotlib.image.AxesImage at 0x7ff644185550>
```

Figure 7: Child playing near water image

VI. LIMITATIONS

The proposed model generates image captions that are limited to the phrases which already exist and cannot incorporate a new combination of words. Since our model is reliant on the training data, it is unable to predict terms that fall outside of its lexicon. It might even fall short of fully describing the image's details. [12]

Additionally, in some cases, the generated captions might not be relevant to the source image. Based on how the model was trained, there is a possibility that the generated captions at times refer to the least significant aspect of the input image while ignoring the major aspects.

VII. CONCLUSION & FUTURE WORK

Our image caption-generating model incorporates CNN that encodes the input image into a compact representation which is then followed by an RNN which generates texts that correspond to the image features learned by the model. As a decoder, the LSTM network produces grammatically accurate sequences of words and phrases that best describe the information contained in the input image. We tested on several images and received fairly well-descriptive captions. We may infer from our results that the source of input photos has a substantial influence on the process of feature extraction and, consequently, has a big influence on the captions created.

This is our first attempt to build an image caption generator. We used a small dataset that is Flicker8k due to our system's computational constraints. Our model can be improved by using a larger dataset such as Flicker30 and MS COCO. By adjusting the batch size, the number of layers, the learning rate, and other parameters, we could also perform hyperparameter tuning.

To prevent overfitting of the model, we might even think about using a cross-validation set. An attention module could also be added to the model architecture for additional improvements.

REFERENCES

[1] Image Caption Generator using Big Data and Machine Learning, International Research Journal of Engineering and Technology (IRJET), Vol.7, 4/20.

[2] Shuang Liu, Image Captioning Based on Deep Neural Networks, MATEC Web of Conferences 232, 01052 (2018) Available: https://doi.org/10.1051/matecconf/201823201052

[3] Saad Alawi, Tareq Abed Mohammed, and Saad Al-Zai, “Understanding of a convolution neural network”, IEEE–2017

[4] Yao, T., Pan, Y., Li, Y., Qiu, Z., & Mei, T. (2017). Boosting image captioning with attributes. In Proceedings of the IEEE International Conference on Computer Vision (pp. 4894-4902).

[5] Haoran Wang, Yue Zhang, and Xiaosheng Yu, “An Overview of Image Caption GenerationMethods”,(CIN-2020)

[6]https://www.kaggle.com/code/ysthehurricane/image-caption-generator-tutoria

[7]https://machinelearningmastery.com/develop-a-deep-learning-caption-generation-model-in-python/

[8]https://medium.com/swlh/automatic-image-captioning-using-deep-learning-5e899c127387

[9]https://keras.io/examples/vision/video_classification/

[10]https://www.kaggle.com/datasets/adityajn105/flickr8k

[11]https://medium.com/@raman.shinde15/image-captioning-with-flickr8k-dataset-bleu-4bcba0b52926

[12] He, Shan & Lu, Yuanyao. (2019). A Modularized Architecture of Multi-Branch Convolutional Neural Network for Image Captioning. Electronics. 8. 1417. 10.3390/electronics8121417.

[13] Gupta, Shitiz & Agnihotri, Shubham & Birla, Deepasha & Lamba, Puneet & Jain, Achin & Vaiyapuri, Thavavel. (2021). Image Caption Generation and Comprehensive Comparison of Image Encoders. Progress in Tourism and Hospitality Research. 4. 42-55. 10.5281/zenodo.5196025.