IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Image Caption Generator
(With real-time captions on images)

Gaurav Chadha[1], Gursha Gosal[2], Priya Jha[3], Rudransh Singh Mahra[4], Megha Kumar[5]

Student[1], Student[2], Student[3], Student[4], Assistant Professor[5]
Department Of Computer Science & Engineering[1]
Delhi Technical Campus, Greater Noida, India[1]

Abstract : Image caption generation has been an area of great interest for researchers in artificial intelligence. Artificial intelligence has played a key importance driving the enhancement of latest technologies and is expected to continue doing so in the future.

Image captioning involves creating a written description of an image by identifying the objects, their characteristics, and the relationships between them in the image. It then generates the caption based on this information.

In this research, we explore different image caption generation models based on deep neural networks. We focus on various feature extraction and encoder models, and evaluate their performance using the Flickr30K dataset, which contains 31783 images. We also apply these models to generate captions for real-time images and compare their accuracy and effectiveness.

The aim of this research is to check the deep learning models and image processing techniques for identifying objects in an image, understanding the relationships between these objects and generating corresponding captions. This is a rapidly expanding area of study in the world of computer vision with numerous advanced applications, such as self-driving vehicles, e-signatures, and assistive technology for the visually impaired.

Keywords: image caption, artificial intelligence, CNN, LSTM.

INTRODUCTION

Image caption generation involves creating written descriptions of real-time scenes and objects, a fundamental goal of computer vision. The ability to automatically generate captions that accurately describe the content of an image using proper grammar can be a challenging task, but it has numerous practical applications. For example, it can assist people who are visually impaired by providing a verbal description of the content of an image. It can also be useful for self-driving vehicles, allowing them to gain a better understanding of their surroundings. This task is particularly difficult because the generated caption must not only describe the important objects in the image, but also convey the relationships between those objects, their attributes, and the actions they are participating in. Additionally, the visual understanding of the image must be expressed in written language or audio form. Therefore, language or audio generation is a necessary component of thorough visual understanding. In this project, we will utilize transfer learning techniques based on attention mechanisms to extract features from images using various CNN architectures and combine these features.

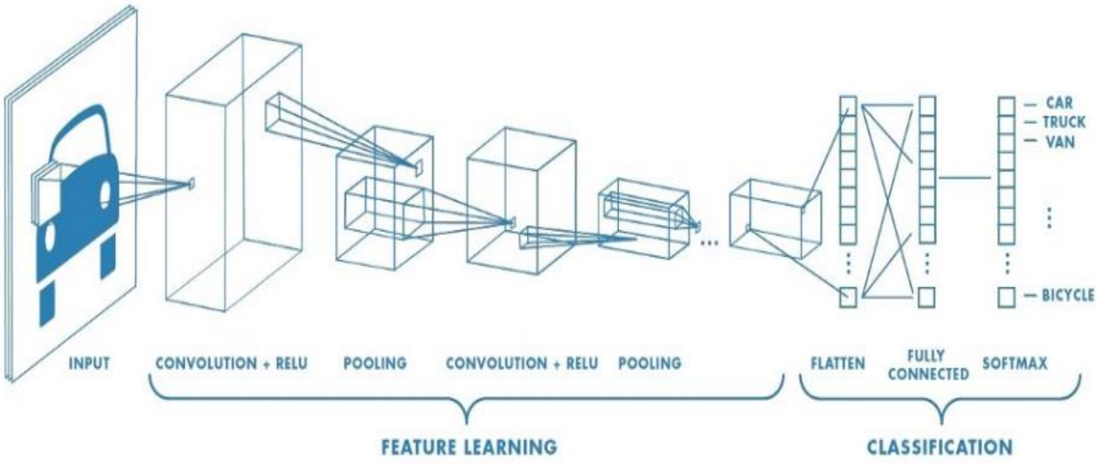


Fig 1: CNN architecture

Image captioning involves generating written descriptions of images. It is a rapidly growing field in the field of computer vision. To perform image captioning, it is necessary to identify the key objects, attributes, and relationships present in an image, as well as generate sentences that are syntactically and semantically correct.

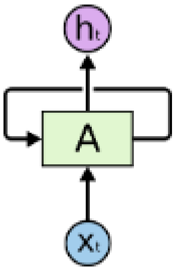


Fig 2: Recurrent Neural Networks loops

Generating accurate captions for images is a significant challenge in artificial intelligence, with a wide range of potential applications, such as aiding in robotic vision and assisting the visually impaired. There is also potential for using these techniques to provide accurate captions for videos in contexts like security systems. Our goal is to develop an optimal system for generating semantically and grammatically correct captions for images. Researchers have been working to improve the accuracy of these predictions, and we have discussed several methods for achieving good results. We have employed deep neural networks and machine learning techniques to build a robust model. We have used the Flickr 30k dataset, which consists of approximately 31,783 sample images with five captions per image, as well as generated captions for real-time images. There are two main phases to this process: feature extraction from the image using convolutional neural networks (CNN), and sentence generation. For the first phase, rather than just detecting the objects in the image, we have adopted a different approach of extracting features from the image, which allows us to identify even subtle differences between similar images. We have used the VGG-16 model (Visual Geometry Group), which is a 16-layer CNN model for object recognition. For the second phase, we need to train our features with the provided captions in the dataset. We are using two architectures for framing sentences from the input images: long LSTM and gGRU. To determine which architecture performs better, we have used the BLEU (Bilingual Evaluation Understudy) score.

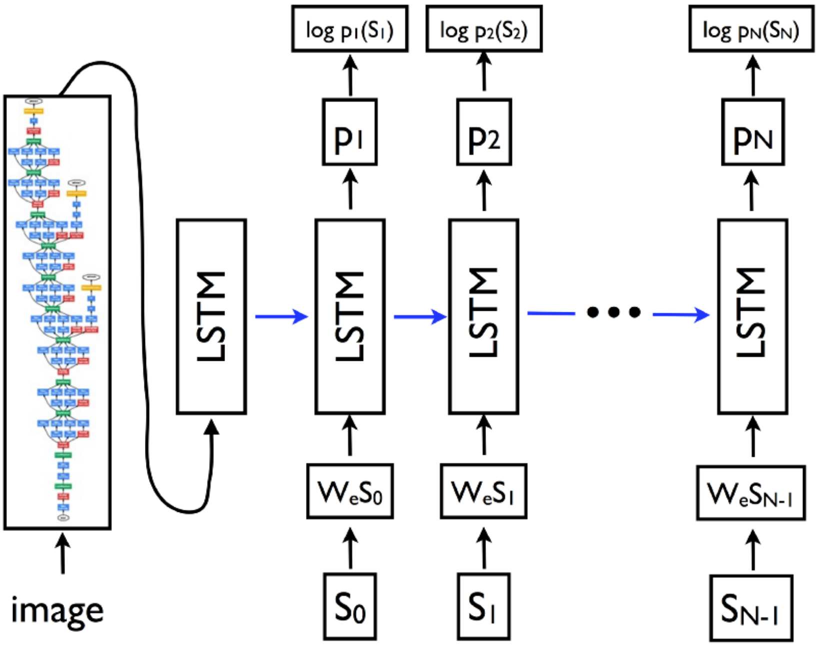


Fig 3 : LSTM Architecture

BACKGROUND STORY

The goal of this project is to generate captions for images by identifying the important objects, attributes, and relationships present in the image and generating sentences that are syntactically and semantically correct. The issue involves a caption generation task that needs a computer vision system to locate and describe important elements in images using natural language.

Image caption generation involves creating a description of an image that consists of a single word, which is a generalization of object detection. Given a set of images and prior knowledge about their content, the task is to identify the appropriate semantic label for the entire image(s).

While this model is effective at identifying the objects in an image, it is not able to describe the relationships between them (e.g., it is a plain image classification model).

COMPARATIVE STUDY

While previous models have been successful at generating captions for images, they have not been able to describe the relationships between the objects depicted in the images (i.e., they are plain image classification models) .

In this project, we have generated captions for real-time images and compared the performance of various feature extraction and encoder models to determine which ones produce the most accurate and effective results.

We have created a generative model that uses a deep recurrent architecture and combines advances in computer vision and machine translation to generate natural language descriptions of images.

PROPOSED METHODOLOGY

The complete system is a combination of five models which optimizes the whole procedure of caption description from an image. The models are

I. NLP / Preprocessing / Tokenization
II. Extraction of Feature Vectors
III. Layering the CNN model
IV. Training the model
V. Testing the model.

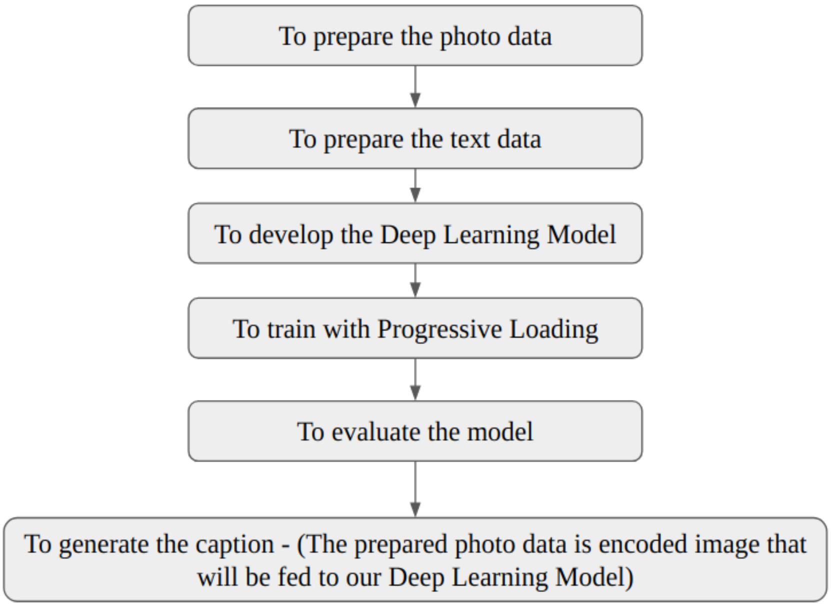

To prepare the photo data
To prepare the text data
To develop the Deep Learning Model
To train with Progressive Loading
To evaluate the model
To generate the caption - (The prepared photo data is encoded image that will be fed to our Deep Learning Model)

Fig 1 : Proposed Methodology Data Flow Diagram

TECHNOLOGY

- Programming:
  - Python 3.11.0
  - Google Colaboratory GPU(Graphics Processing Unit)
  - JavaScript
- Training Phase:
  - Keras API
  - Pickle module
  - Numpy module
  - PIL module
- Testing Phase:
  - IPython module
  - Google. colab module

3.1 NLP / Preprocessing / Tokenization

This particular function loads the file and reads its contents as a string. It then creates a dictionary that associates each image with a list of five captions.

The function performs data cleaning on all the descriptions, such as removing punctuation and converting the text into lowercase. It also creates a vocabulary of all the unique words in the descriptions and a list of pre-processed descriptions, which are saved in a file. The function uses the Keras library's tokenizer function to create tokens from the vocabulary and saves them in a "tokenizer.p" pickle file. The function then maps each word in the vocabulary to a unique index value and stores them in the "descriptions.txt" file.

3.2 Extraction of Feature Vectors

This model is primarily responsible for extracting features from images for use in training. During the training process, the features of the images are input to this model.

The model extracts the features for all images and creates a dictionary that maps image names to their corresponding feature arrays. It then saves this dictionary in a "features.p" pickle file.

3.3 Layering the CNN model

The feature extractor module is responsible for reducing the dimensionality of the features extracted from the image from 2048 to 256 nodes using a dense layer.

The decoder module combines the output from the feature extractor and sequence processor modules and passes it through a dense layer to generate the final prediction. The number of nodes in the final layer is determined by the size of the vocabulary used by the model.

3.4 Training the model

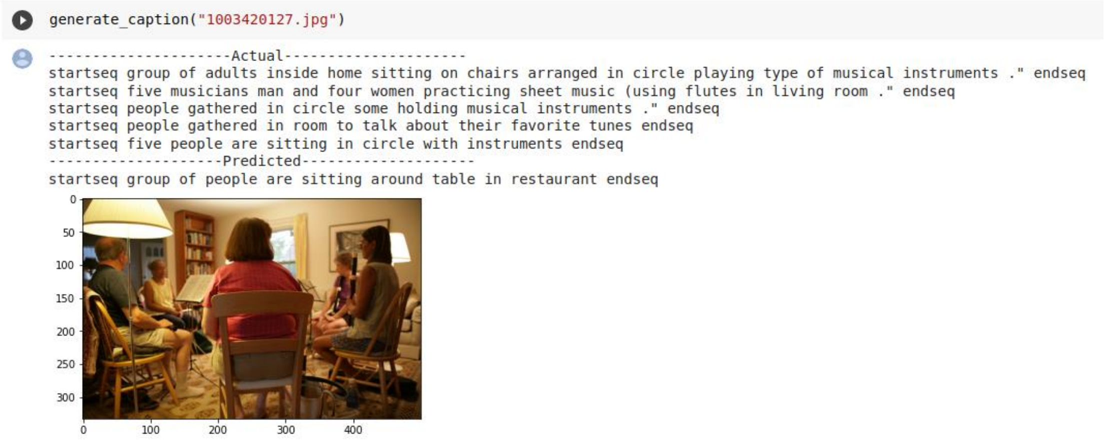
The training model creates a dictionary that maps each photo to its captions and appends the <initial> and <last> identifiers to a particular caption. This allows the model to know the beginning and end of the caption.

In this research, we have employed a long short-term memory (LSTM) layer in our model. This layer helps the model learn how to generate coherent sentences by predicting the word with the highest probability of occurring after a specific word is encountered. This is a key aspect of the model's ability to generate valid sentences.

3.5 Testing the model

After training the model, we created a separate file called "testing_caption_generator.py" that loads the model and generates predictions. The predictions are represented as a sequence of the max length of the index values, "tokenizer.p" pickle file to convert these index values back into the corresponding words.

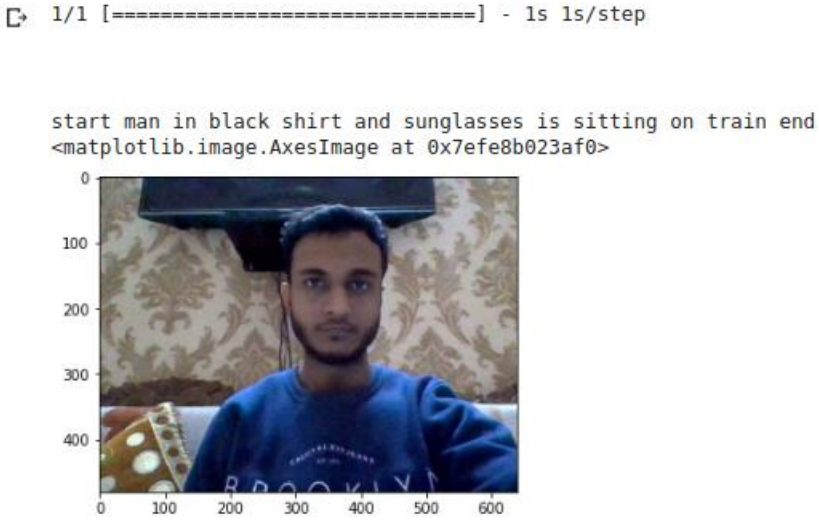
WORKING MODEL



```
generate_caption("1003420127.jpg")
```

```
---------------------Actual---------------------
startseq group of adults inside home sitting on chairs arranged in circle playing type of musical instruments ." endseq
startseq five musicians man and four women practicing sheet music (using flutes in living room ." endseq
startseq people gathered in circle some holding musical instruments ." endseq
startseq people gathered in room to talk about their favorite tunes endseq
startseq five people are sitting in circle with instruments endseq
--------------------Predicted--------------------
startseq group of people are sitting around table in restaurant endseq
```

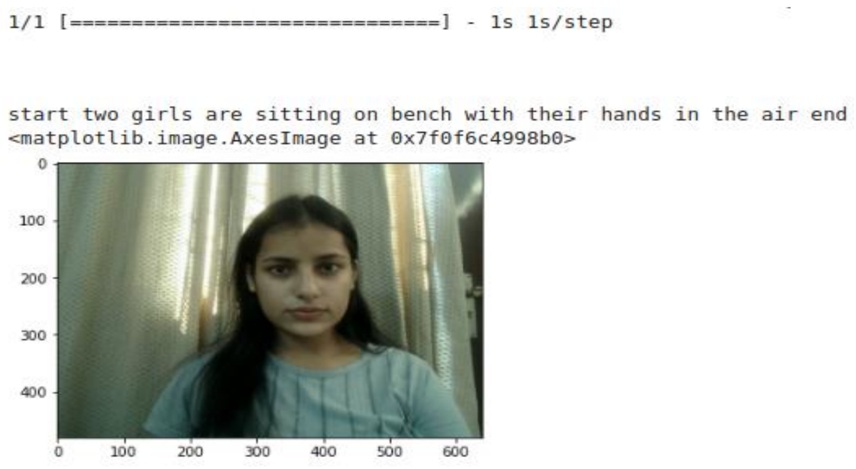
Fig 1: Result from 30k dataset



```
1/1 [==============================] - 1s 1s/step


start man in black shirt and sunglasses is sitting on train end
<matplotlib.image.AxesImage at 0x7efe8b023af0>
```

Fig 2: Result on real time-image-1



```
1/1 [==============================] - 1s 1s/step


start two girls are sitting on bench with their hands in the air end
<matplotlib.image.AxesImage at 0x7f0f6c4998b0>
```

Fig 3: Result on real time-image-2

FUTURE SCOPE

There is potential for further development of the system to make it more user-friendly for researchers. A User Interface could be made for an easy upload and better UX purposes. As seen from applications it is being applied in various industries and in the coming time, these algorithms can be merged with Artificial intelligence to give better and automated results. Speech generation of the description could also be added.

CONCLUSION

- The project was able to meet its objectives, and provide a satisfactory result.
- Assuming the computer system had an Nvidia GPU, it took 43 minutes to extract the feature vectors.
- The main findings of the research were:
    - Some researchers made use of VGG16 instead of the Xception model.
    - Some researchers made use of GRU instead of LSTM.
    - BLEU (Bilingual Evaluation Understudy) Score can be used to check the accuracy.

3.1 Limitations of the System

I. In this project, we aimed to address the major limitation of the CNN image caption generation model, which is its inability to identify relationships among the objects in an image (i.e., it only performs plain image classification).

II. However, the accuracy and the performance of this model are less which could be improved.

III. This project does not have any security concerns which could be included when the user interface and user engagement will be enabled.

REFERENCES

[1]A.Jain.2020. “Flicker30K.”Kaggle. https://www.kaggle.com/datasets/adityajn105/flickr30k

[2] X.Yang. , H.Zhang, J.Cai. 2022. “Auto-Encoding and Distilling Scene Graphs for Image Captioning.” IEEE Transactions https://www.computer.org/csdl/journal/tp/2022/05/09279262/1pg8vG979cs

[3] J.Reimer. 2021. “mountain man made in mexico” Pexels. https://www.pexels.com/photo/cowboy-riding-a-horse-on-the-river-9899960/

[4] D.Sajay. 2019. “Image Caption Generator” Github. https://github.com/dabasajay/Image-Caption-Generator

[5] C.Murray. 2018. “Building an image caption generator with Deep Learning in Tensorflow” Freecodecamp. https://www.freecodecamp.org/news/building-an-image-caption-generator-with-deep-learning-in-tensorflow-a142722e9b1f/

[6] P.Ganesh. 2019. “Type of Convolution Kernels : Simplified” Towards Data Science.https://towardsdatascience.com/types-of-convolution-kernels-simplified-f040cb307c37

[7] F.Chollet. 2018. “A ten-minute introduction to sequence-to-sequence learning in Keras”The Keras Blog. https://blog.keras.io/a-ten-minute-introduction-to-sequence-to-sequence-learning-in-keras.html