International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 2 Feb. 2023, pp: 100-105 www.ijaem.net ISSN: 2395-5252

Generation of Caption for Image Using Deep Learning

Nandaluri Naga Basi Reddy[1], Dr. K. Rajitha[2]
[1]*Student, Mahatma Gandhi Institute Of Technology, Hyderabad, Telangana.*
[2]*Assistant Professor, Mahatma Gandhi Institute of Technology, Hyderabad, Telangana.*

Date of Submission: 01-02-2023 Date of Acceptance: 10-02-2023

ABSTRACT: For this experiment, I utilised CNN and LSTM to determine the image's caption. Large datasets and powerful computers are helpful in the development of models that can create captions for images as deep learning techniques advance. This is what I'll do in this Python-based project where I'll be using CNN and RNN deep learning techniques. To understand the context of a picture and deliver it in English, an image caption generator uses computer vision and natural language processing techniques. I carefully adhered to some of the fundamental ideas and methods used in image captioning for this effort. For the creation of this project, I talked about the Keraslibrary[7], NumPy, and Kaggle notebooks.

I. INTRODUCTION:

We come across numerous images every day from a variety of sources, including the internet, news stories, schematics in documents, and ads. These sites include pictures that visitors must interpret for themselves. Although the majority of photographs lack descriptions, most people can still understand them without them. However, if people require automatic image captions from the machine, it must be able to understand some kind of captions. Many factors make image captioning crucial. Every image on the internet should have a caption to make image searches and indexing more efficient and descriptive. Since researchers have been working on object recognition in photos, it has become obvious that a detailed human-like description is preferable to just listing the names of the items recognised. Natural language descriptions will continue to be a problem to be solved if machines do not think, speak, and act like humans. There are several uses for image captioning in a variety of industries, including biomedicine, business, online search, and the military, among others. Social media platforms like Instagram, Facebook, and others can automatically create captions from photographs.

II. LITERATURE REVIEW:

Recently, image captioning has attracted a lot of interest, particularly in the natural language realm. It may seem far-fetched, but recent advancements in the fields of neural networks, computer vision, and natural language processing have paved the way for accurately describing images, i.e., representing their visually grounded meaning. There is an urgent need for context-based natural language descriptions of images. To do this, we are using cutting-edge methods such as Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and the proper datasets of images and their human-perceived descriptions. We show that our alignment approach works by conducting retrieval tests on datasets like Flicker.

Problem Definition: To create a system that consumers may use that uses CNN and LSTM to automatically generate an image description. A crucial and difficult task is to automatically describe the content of photographs using natural language. Although there has been significant progress in computer vision, it is still a relatively new task to let a computer describe an image that is sent to it in the form of a human. Tasks like object recognition, action classification, image classification, attribute classification, and scene recognition are all possible. Thus, we used the flickr8k dataset and the ResNet50 model, a convolutional neural network (CNN) with 50 layers, to create our model for an image caption generator and it will extract the image's characteristics into a 2048-value vector, while ResNet-50 uses a 224*224 image and assigns 2000 distinct classes.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 2 Feb. 2023, pp: 100-105 www.ijaem.net ISSN: 2395-5252

III. METHODOLOGY AND FRAMEWORK:

A) System Architecture-

An advanced form of artificial neural network known as a convolutional neural network substitutes the mathematical operation known as convolution for generic matrix multiplication in at least one of its layers.

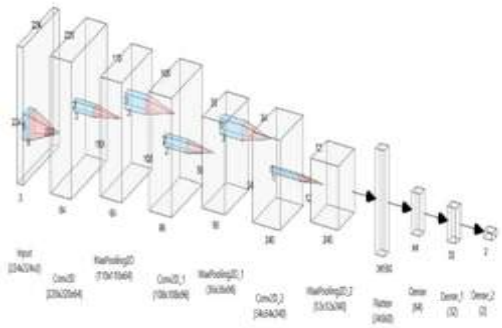


Fig: CNN Architecture

They are employed in image processing and recognition since they were created primarily to process pixel data.A convolutional neural network consists of an input layer, hidden layers, and an output layer. In any feed-forward neural network, any middle layers are called hidden because their inputs and outputs are masked by the activation function and final convolution.

The hidden layers in a convolutional neural network[8] contain convolutional layers. Typically, this involves adding a layer that does a dot product of the input matrix of the layer and the convolution kernel. The activation mechanism for this product, which is often the Frobenius inner product, is frequently ReLU. The convolution procedure develops a feature map as the convolution kernel moves across the input matrix for the layer, adding to the input of the following layer. Following this are further layers like normalising, pooling, and fully connected layers.

B)Algorithms and techniques:

We used the Deep learning Algorithm and Encoder-Decoder language model.

1.Deep learning Algorithm

As it can handle vast volumes of data, deep learning has proven to be a very potent tool. Hidden layer technology is much more popular than conventional methods, particularly for pattern recognition. Convolutional Neural Networks are among the most widely used deep neural network is known as deep learning.

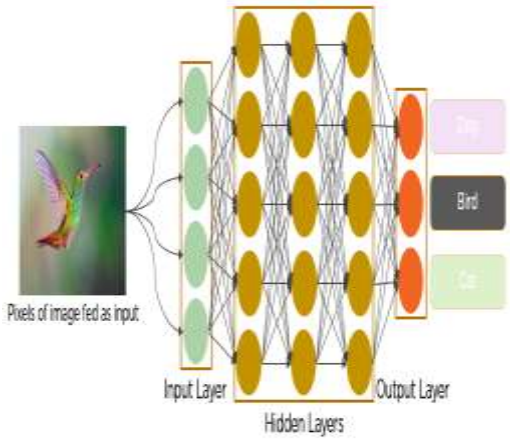


Fig-Concept of Convolutional Neural Networks (CNNs)

2. Encoder Decoder language model

A full input sequence was read and encoded to a fixed-length internal representation using an encoder-decoder architecture.

Then, until the end of the sequence token was reached, a decoder network output words using this internal representation. The encoder and decoder both made use of LSTM networks.

Five deep learning models were used to create the final model. The translations were inferred using a left-to-right beam search.

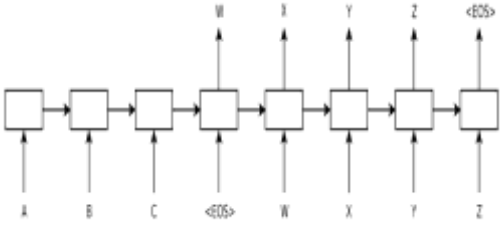


Fig: Encoder-Decoder model for Text Translation

System Design:

A)FLICKR8K DATASET

Each caption gives a precise explanation of the objects and actions shown in the picture. The following characteristics of the dataset make it appropriate for this project.

• Multiple captions mapped for a single image make the model generic and prevent model overfitting A publicly available benchmark dataset for image to sentence description is the Flickr8k dataset. This dataset includes 5 captions per each of the 8091 images.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 2 Feb. 2023, pp: 100-105 www.ijaem.net ISSN: 2395-5252

• Diverse categories of training images can make the image captioning model work for multiple categories of images and therefore can make the model more robust.

B) Image Data Preparation:

To train a deep learning model, the image should be transformed into the appropriate features. To train any image in a deep learning model, feature extraction is a prerequisite. Utilizing the ResNet50 model of the Convolutional Neural Network (CNN), the features are retrieved [6].

Convolutional neural network [10] ResNet-50 has 50 layers total. A pretrained version of the network that has been trained on more than a million photos is available for loading from the ImageNet database. The trained network can categorise photos into 1000 different object categories [9], including several different animals, a mouse, a keyboard, and a pencil. The Resnet-50 will extract the image's features and store them as a vector of 2048 values.

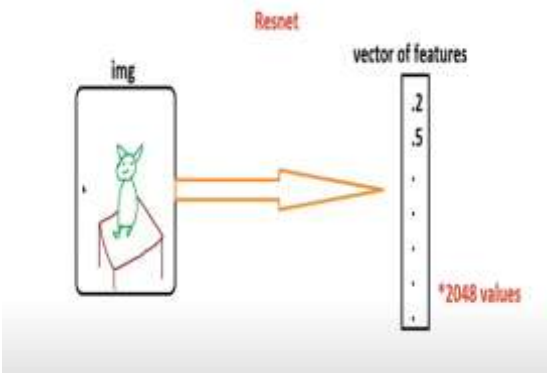


Fig: ResNet50 extract the features from image in the vector of 2048 values

C) Caption Data Preparation

For every image in the Flickr8k collection, numerous descriptions are provided. Each image id is used as a key during the data preparation stage, and a dictionary is used to store the captions that go with it as values.

D) Data cleaning

The raw text of the text dataset must be transformed into a format that can be used by machine learning or deep learning models [12]. Before using the text for the project, the following text cleaning procedures are performed: • Eliminating punctuation. • Elimination of numbers. • Elimination of short words. • Lowercase characters are converted to uppercase characters. Stop words are left in the text data because removing them will make it more difficult to create the grammatically correct caption that is required for this project.

Implementation and Results-

A) Load the data

Downloaded from dataset**:**

• Flickr8k_Dataset – Dataset folder which contains 8091 images.

• Flickr_8k_text – Dataset folder which contains text files and captions of images. The below files will be created by us while making the project.

• Models – It will contain our trained models.

• Descriptions.txt – This text file contains all image names and their captions after pre-processing.

• Testing_caption_generator.py – Python file for generating a caption of any image.

•Training_caption_generator.ipynb – In which we train and build our image caption generator [11].

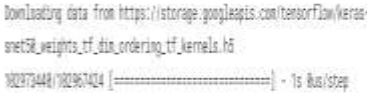
•And we also import the paths of the Images, captions, train and test.

B) Image Pre-processing

1. We need to import some inputs numpy, pandas, cv2, os, and glob to manipulate the images.

And then we'll display the image and the captions to it by using data set.

2. To import ResNet model from keras. And now it will download ResNet50.



3. And now to remove the Dense layer and to keep Averagepooling layer as output layer for that we need to create a new module and take **last = incept.model.layers[-2].output**

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 2 Feb. 2023, pp: 100-105 www.ijaem.net ISSN: 2395-5252

res5c_branch2c (Conv2D)	(None, 7, 7, 2048)	1050624	activation_48[0][0]
bn5c_branch2c (BatchNormalizati	(None, 7, 7, 2048)	8192	res5c_branch2c[0][0]
add_16 (Add)	(None, 7, 7, 2048)	0	bn5c_branch2c[0][0] activation_46[0][0]
activation_49 (Activation)	(None, 7, 7, 2048)	0	add_16[0][0]
global_average_pooling2d_1 (Glo	(None, 2048)	0	activation_49[0][0]

Total params: 23,587,712
Trainable params: 23,534,592
Non-trainable params: 53,120

C) Text Pre-processing

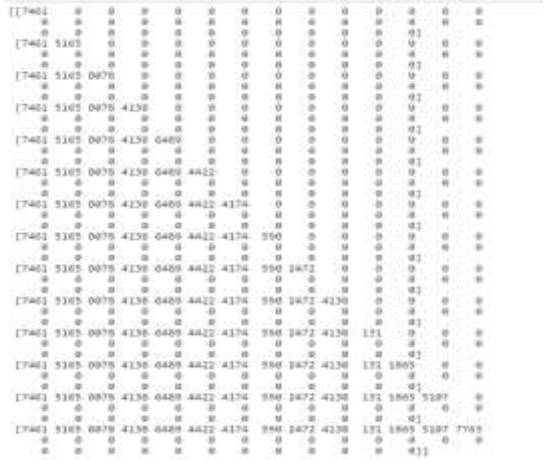
1.Here we add the start and end of sequence at the ends of the captions. This will make easy to us to split the captions for the image code to identify.

	image_id	captions
0	2513260012_03d33305cf.jpg	<start> A black dog is running after a white d...
1	2513260012_03d33305cf.jpg	<start> Black dog chasing brown dog through sn...
2	2513260012_03d33305cf.jpg	<start> Two dogs chase each other across the s...
3	2513260012_03d33305cf.jpg	<start> Two dogs play together in the snow . <...
4	2513260012_03d33305cf.jpg	<start> Two dogs running through a low lying b...

2. Captions stored in sentences

```
['<start> A black dog is running after a white dog in the snow . <end>',
 '<start> Black dog chasing brown dog through snow <end>',
 '<start> Two dogs chase each other across the snowy ground . <end>',
 '<start> Two dogs play together in the snow . <end>',
 '<start> Two dogs running through a low lying body of water . <end>']
```

3. Print the padded sequences



D) Create Vocabulary

An image, and we have encoded [] list i.e., string converted to integer captions.

And now captions dictionary looks like as captions converted to integers and key converted to image code. And we are iterating with captions_dict and after that we iterate with 5 different captions for

E) Build generator function

Formation of an array

```
array([ 7499., 11690.,   342.,     0.,     0.,     0.,     0.,     0.,
           0.,     0.,     0.,     0.,     0.,     0.,     0.,     0.,
           0.,     0.,     0.,     0.,     0.,     0.,     0.,     0.,
           0.,     0.,     0.,     0.,     0.,     0.,     0.,     0.,
           0.,     0.,     0.,     0.])
```

F) MODEL

1. Image model

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 128)	262272
repeat_vector_1 (RepeatVecto	(None, 40, 128)	0

Total params: 262,272
Trainable params: 262,272
Non-trainable params: 0

2. Language model

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 2 Feb. 2023, pp: 100-105 www.ijaem.net ISSN: 2395-5252

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 40, 128)	1056512
lstm_1 (LSTM)	(None, 40, 256)	394240
time_distributed_1 (TimeDist	(None, 40, 128)	32896

Total params: 1,483,648
Trainable params: 1,483,648
Non-trainable params: 0

3.Concatenate the models

Layer (type)	Output Shape	Param #	Connected to
embedding_1_input (InputLayer)	(None, 40)	0	
dense_1_input (InputLayer)	(None, 2048)	0	
embedding_1 (Embedding)	(None, 40, 128)	1056512	embedding_1_input[0][0]
dense_1 (Dense)	(None, 128)	262272	dense_1_input[0][0]
lstm_1 (LSTM)	(None, 40, 256)	394240	embedding_1[0][0]
repeat_vector_1 (RepeatVector)	(None, 40, 128)	0	dense_1[0][0]
time_distributed_1 (TimeDistrib	(None, 40, 128)	32896	lstm_1[0][0]
concatenate_1 (Concatenate)	(None, 40, 256)	0	repeat_vector_1[0][0] time_distributed_1[0][0]
lstm_2 (LSTM)	(None, 40, 128)	197120	concatenate_1[0][0]
lstm_3 (LSTM)	(None, 512)	1312768	lstm_1[0][0]
dense_2 (Dense)	(None, 8254)	4234302	lstm_3[0][0]
activation_50 (Activation)	(None, 8254)	0	dense_2[0][0]

Total params: 7,490,110
Trainable params: 7,490,110
Non-trainable params: 0

4. Model fit

```
25493/25493 [==============================] - 9s 361us/step - loss: 0.2677 - acc: 0.8996
Epoch 195/200
25493/25493 [==============================] - 10s 385us/step - loss: 0.2672 - acc: 0.9004
Epoch 196/200
25493/25493 [==============================] - 10s 374us/step - loss: 0.2622 - acc: 0.9013
Epoch 197/200
25493/25493 [==============================] - 9s 367us/step - loss: 0.1651 - acc: 0.8999
Epoch 198/200
25493/25493 [==============================] - 9s 368us/step - loss: 0.1636 - acc: 0.9023
Epoch 199/200
25493/25493 [==============================] - 10s 383us/step - loss: 0.2645 - acc: 0.9022
Epoch 200/200
25493/25493 [==============================] - 9s 366us/step - loss: 0.2605 - acc: 0.9004
```

IV. CONCLUSION

I have explored deep learning-based picture captioning techniques in this research. I used the roughly 8000-image Flickr 8k dataset, and the text file also has the captions that go with each of the images. I had also processed images and texts, accordingly. Additionally, I had observed how ResNet50 operated in the dataset to generate the encoding. And how the image caption creation issue is dealt with using a recurrent neural network design with encoder-decoder functionality. I have also witnessed how the LSTM decoding process operates. Although deep learning-based image captioning techniques have made significant advances in recent years, a reliable technique that can provide captions of a high calibre for almost all images has not yet been developed. Automatic picture captioning will continue to be a widely researched topic for some time to come with the introduction of novel deep learning network architectures. With more people using social media daily and most of them posting images, the potential for image captioning is very broad in the future. Therefore, they will benefit more from this project.

V. FUTURE SCOPE

Due to the internet's and social media's exponential rise in image content, captioning for photographs has recently become a significant issue. Future study in this area has a huge potential because feature extraction and similarity calculation in images are difficult tasks. By including new picture captioning datasets into the project's training, it will be possible to improve the identification of classes with lesser precision in the future. In order to see if the picture retrieval outcomes improve, this methodology can also be integrated with earlier image retrieval techniques like the histogram, shapes, etc. Additionally, there are many uses for image captioning, including suggestions in editing software, use in virtual assistants, image indexing, for people with visual impairments, on social media, and in several other natural language processing applications.

REFERENCES

[1]. Soheyla Amirian∗, Khaled Rasheed†, Thiab R. Taha‡, Hamid R. Arabnia, “Image Captioning with Generative Adversarial Network”, IEEE,International Conference on Computational Science and Computational Intelligence (CSCI), 2019

[2]. SheshangDegadwala, Dhairya Vyas, Haimanti Biswas, Utsho Chakraborty, SowravSaha, “Image Captioning Using Inception V3 Transfer Learning Model”, Proceedings of the 6th International Conference on Communication and Electronics Systems (ICCES-2021) IEEE Xplore Part Number: CFP21AWO-ART; ISBN: 978-0-7381-1405-7

[3]. Vaishnavi Agrawal, SharivaDhekane, Vibha Vyas, Neha Tuniya, “Image

Caption generator using attention mechanism", IEEE, 2021

[4]. Ansar Hani, NajibaTagougui, MonjiKherallah, "Image caption generation using a deep architecture", International Arab Conference on Information Technology ACIT, 2019.

[5]. Subhash Chand Gupta, Nidhi Raj Singh, Tulsi Sharma, AkshitaTyagi, Rana Majumdar, "Generating Image Captions using Deep Learning and Natural Language Processing", 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO) Amity University, Noida, India. Sep 3-4, 2021

[6]. PranayMathur, Aman gill, Aayush Yadav, Anurag Mishra, Nand Kumar, "Camera2Caption: A Real-Time Image Caption Generator", IEEE, 2017 International Conference on Computational Intelligence in Data Science (ICCIDS)

[7]. Nishanth Behar and Manish Shrivastava, "ResNet50-BasedEffectiveModelfor BreastCancer Classification Using Histopathology Images", Tech Science press, 2021

[8]. K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, et al., "Show, attend and tell: Neural image caption generation with visual attention", Proceedings of the International Conference on Machine Learning (ICML), 2015.

[9]. J. Redmon, S. Divvala, Girshick and A. Farhadi, "You only look once: Unified real-time object detection", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016

[10]. SaadAlbawi, Tareq Abed Mohammed, and Saad Al-Zawi, "Understanding of a convolutional neural network", IEEE – 2017

[11]. Automatic Caption Generation for News Images by Yansong Feng, and Mirella Lapata, IEEE (2013).

[12]. Image Caption Generator Based on Deep Neural Networks by Jianhui Chen, Wenqiang Dong and Minchen Li, ACM (2014).