

IMAGE CAPTION GENERATION AND TRANSLATION

MINOR PROJECT REPORT

BACHELOR OF TECHNOLOGY

Computer Science and Engineering

SUBMITTED BY:-

GUNEET KOHLI (1805172)
JASHANPREET KAUR (1805188)
KARTIKA (1805192)

SUBMITTED TO :-

PROF. MANJOT KAUR GILL

Jan-May, 2021


Scanned with CamScanner

Department of Computer Science and Engineering
GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA

Abstract

Image captioning is a challenging task where computer vision and natural language processing both play a part to generate captions. This technology can be used in many new fields like helping the visually impaired, medical image analysis, geospatial image analysis etc.

Image caption generator is a model which generates caption based on the features present in the input image.

The basic working of the project is that the features are extracted from the images using a pre-trained VGG16 model and then fed to the LSTM model along with the captions to train. The trained model is then capable of generating captions for any images that are fed to it.

Bottom Up and Top down are two main approaches to Image captioning Bottom Up approach generate items observed in an image, and then attempt to combine the items identified into a caption. Top down approach attempts to generate a semantic representation of an image that is then decoded into a caption using various architectures, such as recurrent neural networks. The top down approach follows in the footsteps of recent advances in statistical machine translation, and the state-of-the-art models mostly adopt the top-down approach.

Top down approach is used in our model, one of the successful implementation of top down approach is a model that uses a deep convolutional neural network to generate a vectorized representation of an image that then feed into a Long-Short-Term Memory (LSTM) network, which then generates captions. Experiments on several datasets show the accuracy of the model and the fluency of the language it learns solely from image description.

CNNs(Convolutional Neural networks) can produce a rich representation of the input image by embedding it to a fixed-length vector, such that this representation can be used for a variety of vision tasks .So CNN will be used as an image “encoder”, by first pre-training it for an image classification task and using last hidden layer as an input to the RNN decoder that generates sentences. This model can be referred to as Neural Image Caption. The image features can also be extracted from Xception or VGG-16 or Inception V3 which are CNN models trained on imagenet dataset and then we can feed these features into the LSTM model which will be responsible for generating the image captions. Trained model is evaluated using BLEU Scores

ACKNOWLEDGEMENT

I/WE are highly grateful to Dr. Sehijpal Singh, Principal, Guru Nanak Dev Engineering College (GNDEC) Ludhiana, for providing this opportunity to carry out the minor project work on Image caption generation and translation.

The constant guidance and encouragement received from Dr. Parminder Singh H.O.D. CSE Department, GNDEC Ludhiana has been of great help in carrying out the project work and acknowledgement with reverential thanks.

I/WE would like to express a deep sense of gratitude and thanks profusely to Prof. Manjot Kaur Gill, without her wise counsel and able guidance, it would have been impossible to complete the project in this manner.

I/WE express gratitude to other faculty members of the computer science and engineering department of GNDEC for their intellectual support throughout the course of this work.

Finally, I/WE are indebted to all whosoever have contributed in this report work

Kartika

Guneet Kohli

Jashanpreet Kaur

LIST OF FIGURES

Fig. No.	Figure Description	Page No.
4.1	Flow Chart	13
4.2	Trained Model of LSTM Network	14
5.1	Organization of files	15
6.1	Some bad captions with BLEU Scores <0.2	16
6.2	Some good captions with BLEU Scores >0.7	17
7.1	Captions associated to images	18
7.2	LSTM Model Summary	19
7.3	Classified Images	20

TABLE OF CONTENTS

Contents	Page No.
Abstract	2
Acknowledgement	3
List Of Figures	4
Table of Contents	5
Introduction	6-7
Objectives	8
System Requirements	9
Software Requirement Analysis	10-11
Software Design	12-14
Coding/Core Modules	15
Performance Of Model	16-17
Output Screens	18-20
Conclusion	21
Future Scope	22
References	23

1. INTRODUCTION

Image captioning is a challenging task where computer vision and natural language processing both play a part to generate captions. This technology can be used in many new fields like helping the visually impaired, medical image analysis, geospatial image analysis etc.

In Image Caption Generation concepts of Computer Vision and Natural Language Processing are applied to recognize the context of image and describe the context of Image in some Natural Language like English and using Machine Translation and NaturalLanguage Processing the caption can be described in domain language.

In this project our objective will be to train the model to maximize the likelihood of the target description sentence for a given training image.

Use cases of Image Caption Generation

- Some detailed use cases would be like an visually impaired person taking a picture from his phone and then the caption generator will turn the caption to speech for him to understand.
- Advertising industry tries to generate captions automatically without the need to make them separately during production and sales.
- Doctors can use this technology to find tumors or some defects in the images or used by people for understanding geospatial images where they can find out more details about the terrain.

1.1 DATASET USED AND PROCESSING OF TRAINING DATA

For training, the FLICKR_8K dataset will be used. It is a labeled dataset consisting of 8000 photos with 5 captions for each photo written by different people for each image. It includes images obtained from the Flickr website. The images and descriptions will be loaded, and then define the model and the learning process. Finally, captions for new images will be generated.

There are also other big datasets like Flickr_30K and MSCOCO dataset but it can take weeks just to train the network so we will be using a small Flickr_8k dataset. The advantage of a huge dataset is that we can build better models.

For processing the training set, all the images will be resized to standard size. Encode the images, and convert the PIL image to numpy array, and then the numpy array to 2D. Perform all the preprocessing needed by VGG16 or Xception or InceptionV3, and call any of them to extract as smaller feature set, thus getting the encoding vector for the image.

It will be shaped to the correct form, so that it is accepted by the LSTM captioning network, looping over every JPG image. Word removal of the words which occur less than 10 times is done, as these words mislead the neural network, and the data is significantly reduced by approximately 7 times.

Concept of look-up tables will be used, one for converting index numbers to words and another for converting actual words to index numbers.

OBJECTIVES

1. To clean and process the Flickr_8k data set so that it can be trained and features could be extracted from it.
2. To learn and understand the concepts of VGG16 ,Inception V3 and Xception and to learn how to use them for feature extraction.
3. To analyse a given image using VGG 16(VGG16 (also called OxfordNet) is a convolutional neural network architecture named after the Visual Geometry Group fromOxford, who developed it. It was used to win the ILSVRC (ImageNet) competition in2014) or Inception v3 or Xception for feature extraction.
4. To learn the concepts of CNN and LSTM model and build a working model of Image caption generator by implementing CNN with LSTM.
5. To implement CNN, LSTM or other deep learning models to model the extracted features and generate captions from images.
6. To train the model to maximize the likelihood of the target description sentence for a given training image.

2. SYSTEM REQUIREMENTS

- **Software**
 - Jupyter Notebook
 - Google Colab
 - Python --version 3
 - Tensorflow Library
 - Keras
 - Pillow
 - Numpy
 - 8 Gb RAM
 - i5 Processor
- **Hardware**
 - OS(linux and windows)
 - Intel core i5 Processor

3. Software Requirement Analysis

Image caption generator is a model which generates caption based on the features present in the input image.

The basic working of the project is that the features are extracted from the images using a pre-trained VGG16 model and then fed to the LSTM model along with the captions to train. The trained model is then capable of generating captions for any images that are fed to it.

Various Modules :

3.1. Cleaning Caption Data :

This is the first step of data pre-processing. The captions contain regular expressions, numbers and other stop words which need to be cleaned before they are fed to the model for further training. The cleaning part involves removing punctuations, single character and numerical values. After cleaning we try to figure out the top 50 and least 50 words in our dataset.

3.2. Adding start and end sequence to the captions

Start and end sequences need to be added to the captions because the captions vary in length for each image and the model has to understand the start and the end.

3.3. Extracting features from images

- After dealing with the captions we then go ahead with processing the images. For this we make use of the pretrained VGG-16 weights.
- Instead of using this pre-trained model for image classification as it was intended to be used. We just use it for extracting the features from the images. In order to do that we need to get rid of the last output layer from the model. The model then generates 4096 features from taking images of size (224,224,3).

3.4. Viewing similar images

When the VGG-16 model finishes extracting features from all the images from the dataset, similar images from the clusters are displayed together to see if the VGG-16 model has extracted the features correctly and we are able to see them together.

3.5. Merging the caption with the respective images

- The next step involves merging the captions with the respective images so that they can be used for training. Here we are only taking the first caption of each image from the dataset as it becomes complicated to train with all 5 of them.
- Then we have to tokenize all the captions before feeding it to the model.

3.6. Splitting the data for training and testing

The tokenized captions along with the image data are split into training, test and validation sets as required and are then pre-processed as required for the input for the model.

3.7. Building the LSTM model

The LSTM model has been used because it takes into consideration the state of the previous cell's output and the present cell's input for the current output. This is useful while generating the captions for the images.

The step involves building the LSTM model with two or three input layers and one output layer where the captions are generated. The model can be trained with various numbers of nodes and layers. We start with 256 and try out with 512 and 1024. Various hyperparameters are used to tune the model to generate acceptable captions

3.8. Predicting on the test dataset and evaluating using BLEU scores

After the model is trained, it is tested on a test dataset to see how it performs on caption generation for just 5 images. If the captions are acceptable then captions are generated for the whole test data.

4. SOFTWARE DESIGN

Bottom Up and Top down are two main approaches to Image captioning Bottom Up approach generate items observed in an image, and then attempt to combine the items identified into a caption. Top down approach attempts to generate a semantic representation of an image that is then decoded into a caption using various architectures, such as recurrent neural networks. The top down approach follows in the footsteps of recent advances in statistical machine translation, and the state-of-the-art models mostly adopt the top-down approach.

Top down approach will be used in our model ,one of the successful implementation of top down approach is a model that uses a deep convolutional neural network to generate a vectorized representation of an image that then feed into a Long-Short-Term Memory (LSTM) network, which then generates captions. Experiments on several datasets show the accuracy of the model and the fluency of the language it learns solely from image description.

CNNs(Convolutional Neural networks) can produce a rich representation of the input image by embedding it to a fixed-length vector, such that this representation can be used for a variety of vision tasks .So CNN will be used as an image “encoder”, by first pre-training it for an image classification task and using last hidden layer as an input to the RNN decoder that generates sentences. This model can be referred to as Neural Image Caption. The image features can also be extracted from Xception or VGG-16 or Inception V3 which are CNN models trained on imagenet dataset and then we can feed these features into the LSTM model which will be responsible for generating the image captions.

FLOW CHART:-

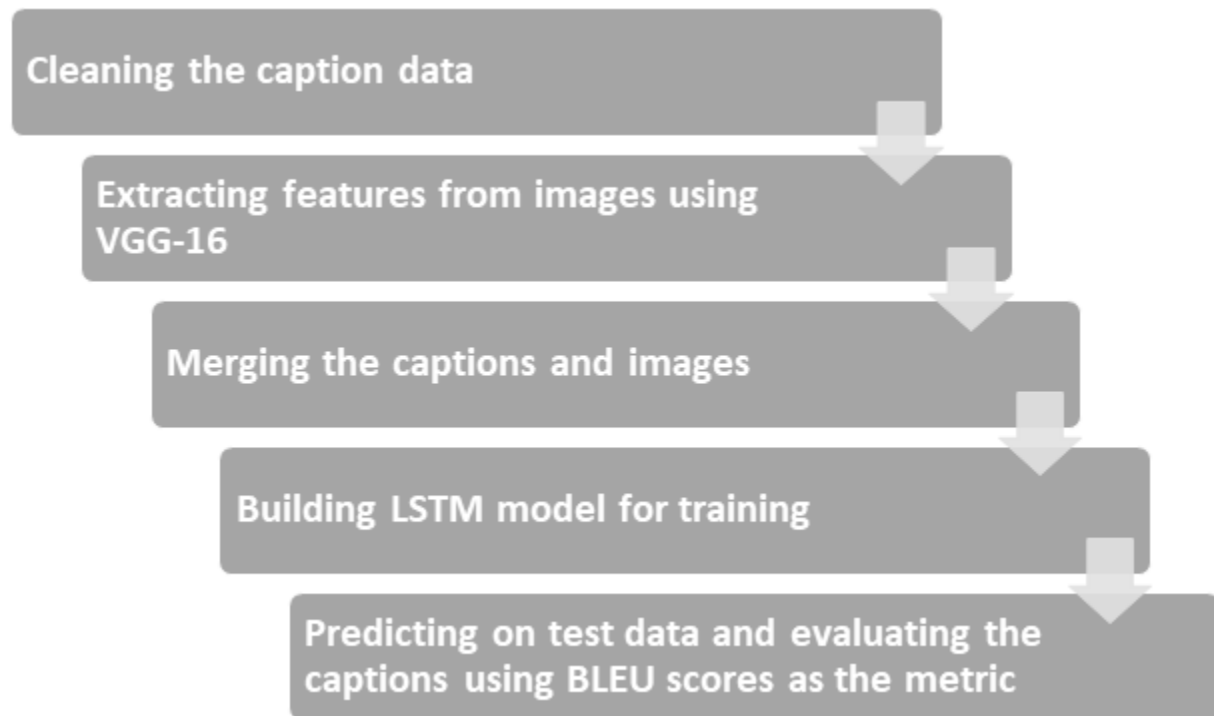


Fig 4.1 Flow Chart

The Image captioning model has been implemented using the Sequential API of keras. It consists of three components:

- a) An encoder CNN model: A pre-trained CNN is used to encode an image to its features. In this implementation VGG16 model is used as encoder and with its pretrained weights loaded. The last softmax layer of VGG16 is removed and the vector of dimension (4096,) is obtained from the second last layer.
The image model takes the (4096,) dimension encoded image vector as input.
- b) A word embedding model: Since the number of unique words can be large, a one hot encoding of the words is not a good idea. An embedding model is trained that takes a word and outputs an embedding vector of dimension (1, 128).
Pre-trained word embeddings can also be used.
- c) A decoder RNN model: A LSTM network has been employed for the task of generating captions. It takes the image vector and partial captions at the current timestep and input and generates the next most probable word as output.

The overall architecture of the model is described by the following picture. It also shows the input and output dimension of each layer in the model.

TRAINED MODEL :-

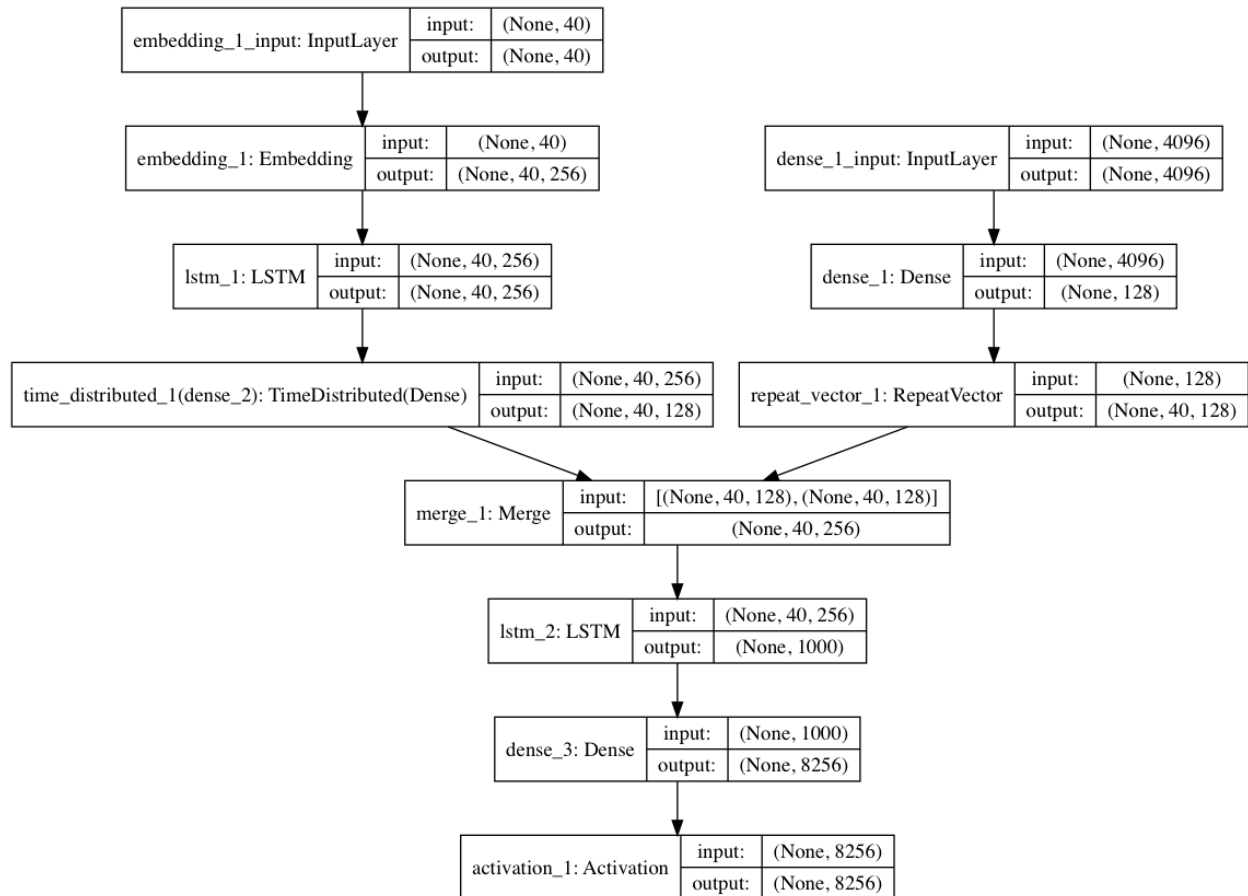


Fig 4.2 Trained Model of LSTM Network

5. CODING/CORE MODEL

Various files used in the project are :

1.Flickr_Data

1.1 flickr8ktextfiles

This includes files :- flickr_8k_train_dataset.txt and flickr_8k_val_dataset.txt

1.2 Flickr_TextData

This includes files :-

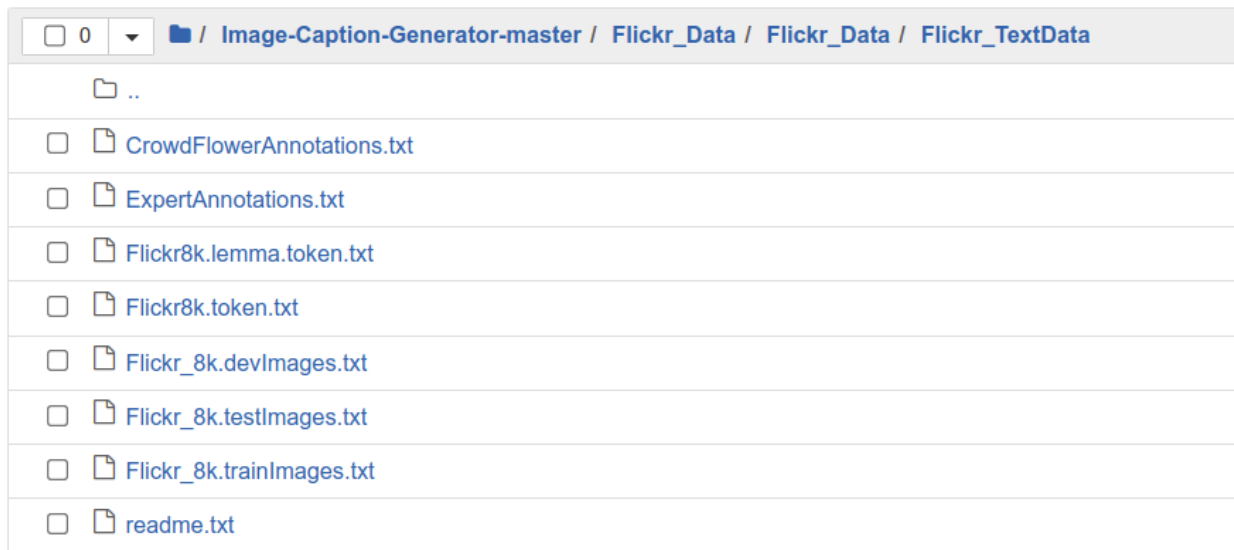


Fig 5.1 Organisation of Files

1.3 Images :

This folder Includes 8k images

2 . Image Captioning 8k .ipynb :

This is the jupyter notebook where the model is constructed

3 . captions.txt :-

This txt file includes 5 captions each corresponding to each image

4. model_weights.h5:-

This file includes pretrained weights for the model .

6. PERFORMANCE OF THE MODEL SO FAR

The model has been trained for 5 epochs which lowers down the loss to 2.6465. With a larger dataset, it might be needed to run the model for at least 5 more epochs. With the current training on the Flickr 8k dataset, running a test on the 1000 test images results in, BLEU = ~ 0.57 .

Some captions generated are as follows:

Bad Caption






	<p>true: little girl covered in paint sits in front of painted rainbow with her hands in bowl</p> <p>pred: two children are sitting on the street</p> <p>BLEU: 0</p>
	<p>true: collage of one person climbing cliff</p> <p>pred: couple at the background</p> <p>BLEU: 0</p>
	<p>true: couple and an infant being held by the male sitting next to pond with near by stroller</p> <p>pred: man is sitting on the street</p> <p>BLEU: 0.121482336484</p>
	<p>true: black dog carries green toy in his mouth as he walks through the grass</p> <p>pred: black dog is shaking the water</p> <p>BLEU: 0.148231563964</p>
	<p>true: black dog and brown dog are jumping up to catch red toy</p> <p>pred: brown dog is running in the grass</p> <p>BLEU: 0.228683500859</p>

Fig 6.1 Some bad captions with BLEU score < 0.2

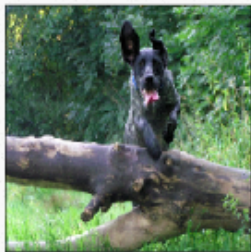
Good Caption



true: black dog and spotted dog are fighting

pred: black and white dog is running through the grass

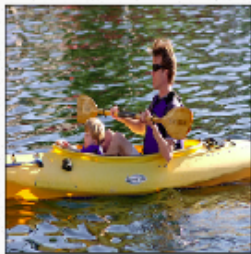
BLEU: 0.759835685652



true: black dog leaps over log

pred: black and white dog is running over the grass

BLEU: 0.759835685652



true: man and baby are in yellow kayak on water

pred: man in blue shirt is sitting in the water

BLEU: 0.759835685652



true: white and black dog and brown dog in sandy terrain

pred: black and white dog is running along the beach

BLEU: 0.730633242659



true: child with helmet on his head rides bike

pred: man in red outfit riding his bike on the road

BLEU: 0.740082804492

Fig 6.2 Some Good Caption with BLEU Score >0.7

7. OUTPUT SCREENS

Jupyter notebook is used to display results:-



startseq boy in red shirt and red jacket is standing on the street endseq



startseq black dog is running in the air in the grass endseq



startseq dog is running through the snow endseq



startseq boy in black jacket is skiing in the snow endseq



startseq group of people are playing on the park endseq

Fig 7.1 Captions associated to images

4476

Layer (type)	Output Shape	Param #	Connected to
input_4 (InputLayer)	(None, 30)	0	
embedding_1 (Embedding)	(None, 30, 64)	286464	input_4[0][0]
CaptionFeature (LSTM)	(None, 30, 256)	328704	embedding_1[0][0]
dropout_1 (Dropout)	(None, 30, 256)	0	CaptionFeature[0][0]
input_3 (InputLayer)	(None, 4096)	0	
CaptionFeature2 (LSTM)	(None, 256)	525312	dropout_1[0][0]
ImageFeature (Dense)	(None, 256)	1048832	input_3[0][0]
add_1 (Add)	(None, 256)	0	CaptionFeature2[0][0] ImageFeature[0][0]
dense_1 (Dense)	(None, 256)	65792	add_1[0][0]
dense_2 (Dense)	(None, 4476)	1150332	dense_1[0][0]
Total params: 3,405,436			
Trainable params: 3,405,436			
Non-trainable params: 0			
None			

Fig 7.2 LSTM Model Summary

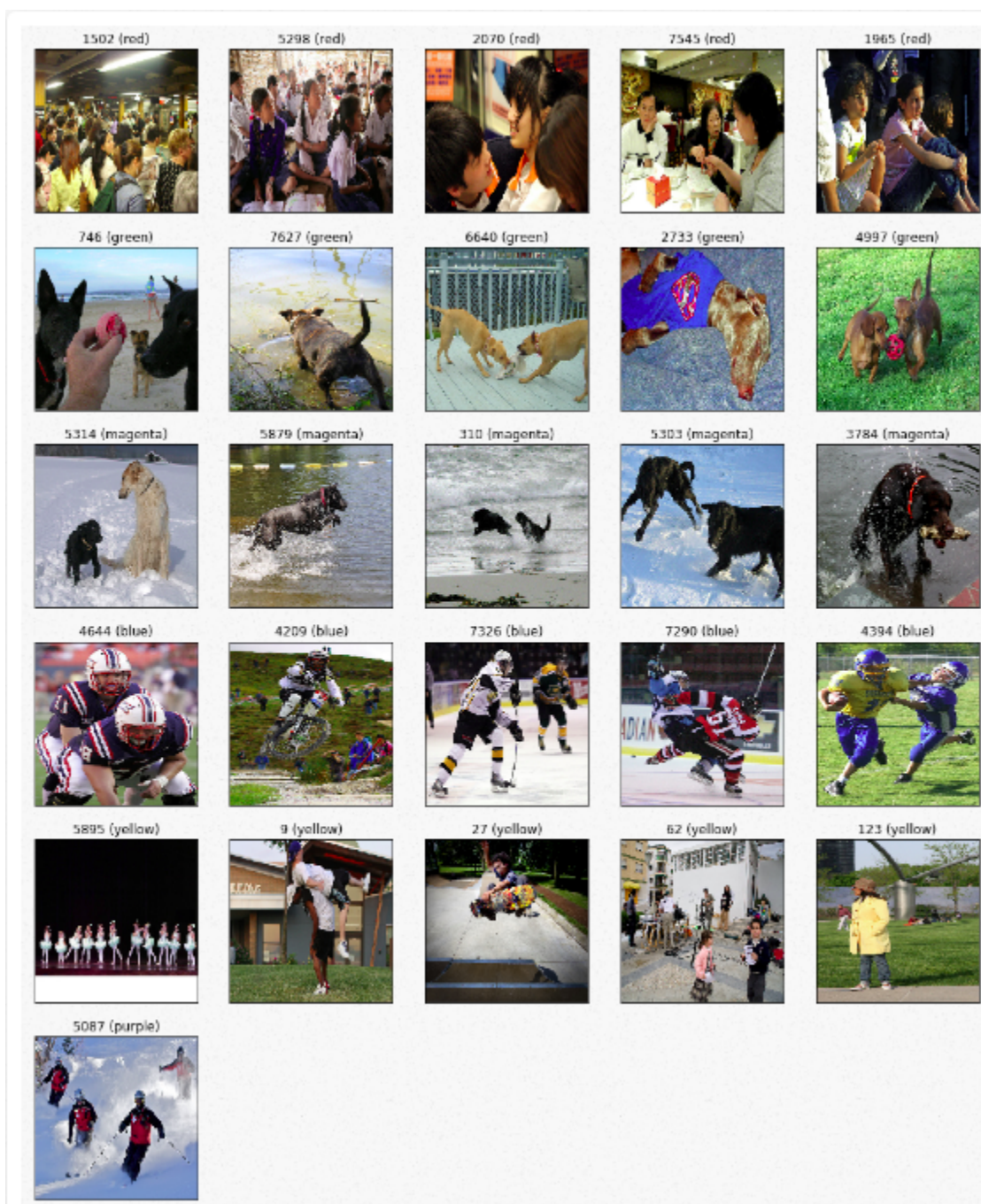


Fig 7.3 Classified images

8.1 CONCLUSION

Image Caption Generation is a Challenging Task and it involves both machine learning and natural language processing concepts.

In this project work , the model that can generate captions for a given image is developed .We had used VGG 16 and LSTM network in our Model . Both Good and Bad captions are generated. Model is evaluated on BLEU Score .

Flickr_8k dataset was used in this project , we processed the dataset using natural language processing techniques. After dealing with the captions we then go ahead with processing the images. For this we make use of the pretrained VGG-16 weights.Instead of using this pre-trained model for image classification as it was intended to be used. We just use it for extracting the features from the images. In order to do that we need to get rid of the last output layer from the model. The model then generates 4096 features from taking images of size (224,224,3).

The next step involves merging the captions with the respective images so that they can be used for training. Here we are only taking the first caption of each image from the dataset as it becomes complicated to train with all 5 of them.

The LSTM model has been used because it takes into consideration the state of the previous cell's output and the present cell's input for the current output. This is useful while generating the captions for the images.

Trained model is evaluated using BLEU Scores.

8.2 FUTURE SCOPE

This project is having great future scope . The captions generated can be translated into domain language so that people using regional language can also interpret them

We can use other criteria for evaluating the model other than BLEU Score to improve the model. Larger Datasets could be used to train the model for efficiency like MSCOCO, flickr_30k. This model could be converted to a web interface.

Image caption can be applied to image retrieval, video caption, and video movement and the variety of image caption systems available today, experimental results show that this task still has better performance systems and improvement. It mainly faces the following three challenges: first, how to generate complete natural language sentences like a human being; second, how to make the generated sentence grammatically correct; and third, how to make the caption semantics as clear as possible and consistent with the given image content. For future work, we propose the following four possible improvements:

- (1) An image is often rich in content. The model should be able to generate description sentences corresponding to multiple main objects for images with multiple target objects, instead of just describing a single target object.
- (2) For corpus description languages of different languages, a general image description system capable of handling multiple languages could be developed by machine translation.
- (3) Evaluating the result of natural language generation systems is a difficult problem. The best way to evaluate the quality of automatically generated texts is subjective assessment by linguists, which is hard to achieve. In order to improve system performance, the evaluation indicators should be optimized to make them more in line with human experts' assessments.
- (4) A very real problem is the speed of training, testing, and generating sentences for the model should be optimized to improve performance as the model takes a long time for training, it took almost 3 hrs for flick_8k dataset .

9. REFERENCES

- [1] Oriol Vinyals, Alexander Toshev, Samy Bengio, Dumitru Erhan. Show and Tell: A Neural Image Caption Generator
- [2] Cyrus Rashtchian, Peter Young, Micah Hodosh, and Julia Hockenmaier. Collecting Image Annotations Using Amazon's Mechanical Turk. In Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk.
- [3] Sharma.G, Kalena.P, et.al, “Visual Image Caption Generator Using Deep Learning” 2019 , 2nd International Conference on Advances in Science & Technology (ICAST-2019) K. J. Somaiya Institute of Engineering & Information Technology, University of Mumbai, Maharashtra, India .
- [4] Kunjukuttan.A,et.al The IIT Bombay English-Hindi Parallel Corpus” 2017
- [5] Chen.J, Dong.W , et.al “Image Caption Generator Based On Deep Neural Networks” 2014 ,International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 9 (2018) pp. 7239-7242 © Research India Publications.
- [6] Josan.G, Lehal.G ,” Direct Approach for Machine Translation from Punjabi to Hindi” 2012,CSI Journal of Computing,Vol 1

Evaluated by: Er. Manjot Kaur Gill