IMAGE CAPTIONING USING CNN AND LSTM

Major project report submitted in partial fulfillment of the requirement for the degree of Bachelor of Technology

in

Computer Science and Engineering/Information Technology

By

ASHIMA PAL (191454)

UNDER THE SUPERVISION OF

Dr. Nafis Uddin Khan

&

Dr. Pradeep Kumar Gupta

to

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology, Waknaghat, Solan-173234, Himachal Pradesh

DECLARATION

I hereby declare that this project has been done by me under the supervision of (Dr Nafis Uddin Khan,Assistant Professor[SG] Deptt. of ECE and Dr. Pradeep Kumar Gupta, Associate Professor, Deptt. of CSE & IT), Jaypee University of Information Technology. I also declare that neither this Project nor any part of this Project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

Dr. Nafis Uddin Khan

Assistant Professor [SG]

Department of Electronics and Communication

Jaypee University of Information Technology

Dr. Pradeep Kumar Gupta

Associate Professor

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology

Submitted by:

Ashima Pal - 191454

Computer Science & Engineering Department Jaypee University of Information Technology

CERTIFICATE

This is to certify that the work which is being presented in the Project report titled “**Image Captioning using CNN and LSTM**” in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science & Engineering** and submitted to the Department of Computer Science & Engineering, Jaypee University of Information Technology, Waknaghat is an authentic record of work carried out by “Ashima Pal (191454)” during the period from February 2023 to May 2023 under the supervision of Dr. Nafis Uddin Khan, Department of Electronics and Communication and Dr. Pradeep Kumar Gupta, Department of Computer Science and Engineering, Jaypee University of Information Technology, Waknaghat.

(Student Signature)

Ashima Pal, 191454

The above statement made is correct to the best of my knowledge.

(Supervisor Signature)

Dr. Nafis Uddin Khan

Assistant Professor[SG]

ECE

(Supervisor Signature)

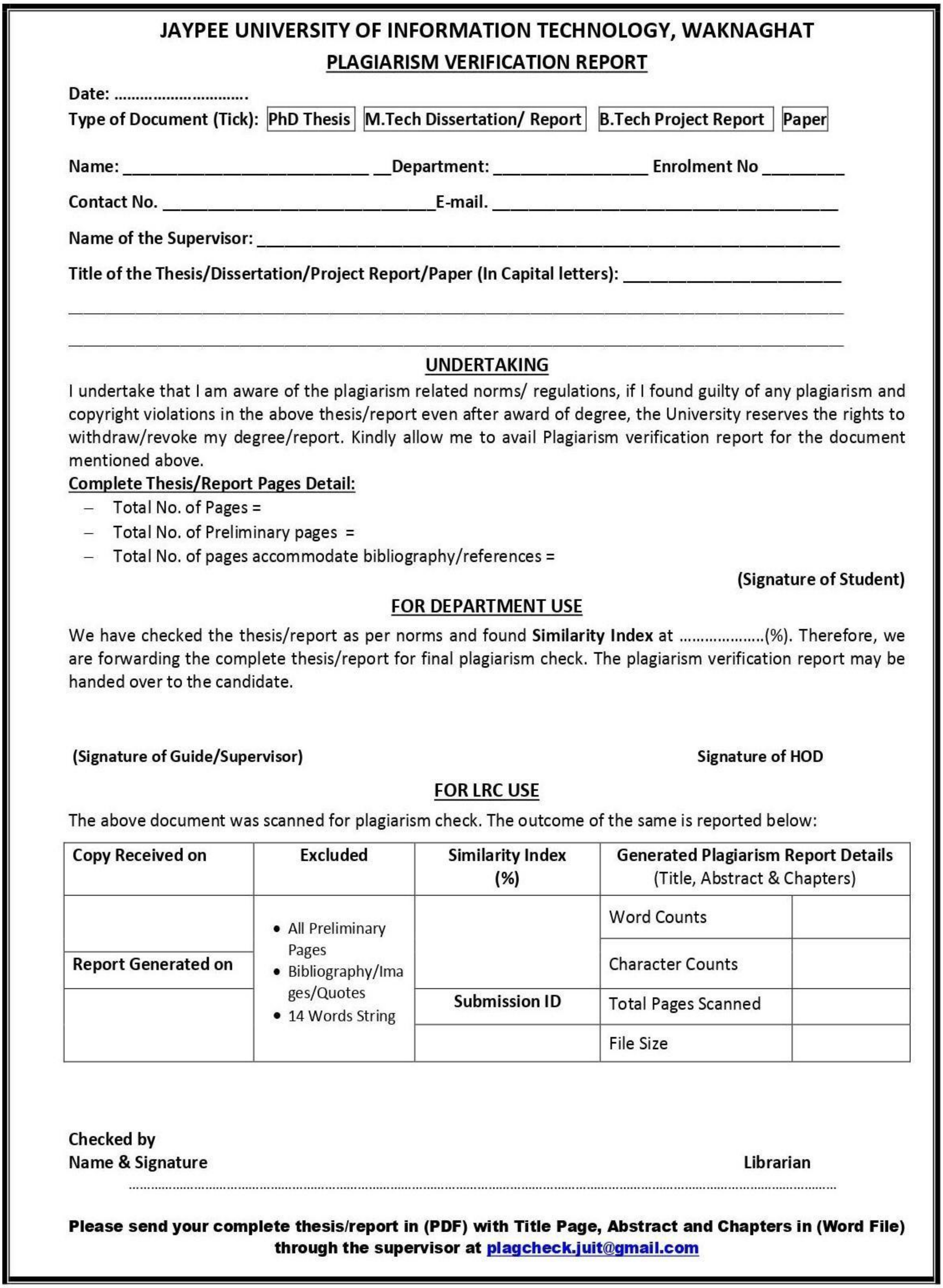
Dr. Pradeep Kumar Gupta

Associate Professor

CSE

Jaypee University of Information Technology, Waknaghat

PLAGIARISM CERTIFICATE



JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date: ………………………….

Type of Document (Tick): □ PhD Thesis □ M.Tech Dissertation/ Report □ B.Tech Project Report □ Paper

Name: __________________________ __Department: ________________ Enrolment No _________

Contact No. _____________________________E-mail. _____________________________________

Name of the Supervisor: ____

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): _______________________

____

____

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages =
- Total No. of Preliminary pages =
- Total No. of pages accommodate bibliography/references =

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at ………………..(%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor) **Signature of HOD**

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

Copy Received on	**Excluded**	**Similarity Index (%)**	**Generated Plagiarism Report Details** (Title, Abstract & Chapters)	
	• All Preliminary Pages • Bibliography/Images/Quotes • 14 Words String		Word Counts	
Report Generated on			Character Counts	
		Submission ID	Total Pages Scanned	
			File Size	

Checked by
Name & Signature **Librarian**

……

Please send your complete thesis/report in (PDF) with Title Page, Abstract and Chapters in (Word File) through the supervisor at plagcheck.juit@gmail.com

ACKNOWLEDGEMENT

First, I express my gratitude to God who provided me with the courage and fortitude to complete the project.

I am grateful and wish my profound indebtedness to Supervisor Dr. Nafis Uddin Khan, Assistant Professor[SG], Department of ECE and Dr. Pradeep Kumar Gupta, Associate Professor, Department of CSE Jaypee University of Information Technology, Wakhnaghat. Deep Knowledge & keen interest of my supervisors in the field of "Artificial Intelligence" to carry out this Project. Their endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this Project.

I would like to express my heartiest gratitude to Dr. Nafis Uddin Khan, Department of ECE and Dr. Pradeep Kumar Gupta, Department of CSE, for their kind help to finish my Project.

I would also generously welcome each one of those individuals who have helped me straightforwardly or in a roundabout way in making this project a success. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, I must acknowledge with due respect the constant support and patience of my parents.

Ashima Pal
191454

TABLE OF CONTENTS

DECLARATION i

CERTIFICATE ii

PLAGIARISM CERTIFICATE iii

ACKNOWLEDGEMENT iv

TABLE OF CONTENTS v

LIST OF FIGURES vi

LIST OF ABBREVIATIONS vii

ABSTRACT viii

Chapter 01: INTRODUCTION 1

1.1 INTRODUCTION 1

1.2 PROBLEM STATEMENT 3

1.3 OBJECTIVES 3

1.4 METHODOLOGY 4

1.5 ORGANIZATION 5

1.5.1 Python 6

1.5.2 NumPy “Numerical Python” 6

1.5.3 Pandas (Python data analysis package) 7

1.5.4 OS 7

1.5.5 Tensorflow 7

1.5.6 Warnings 8

1.5.7 Matplotlib 8

1.5.8 Seaborn 8

Chapter 02: LITERATURE SURVEY 9

2.1 FEASIBILITY STUDY 9

2.2 LITERATURE SURVEY 11

Chapter 03: SYSTEM DEVELOPMENT 21

3.1 ANALYSIS 21

3.2 DESIGN OF THE PROJECT 21

3.2.1 Dataset 22

3.2.2 Flowchart of the Major Project 23

3.3 TRAINING AND TESTING OF MODELS 22

3.3.1 Modules of the Proposed Model 25

3.3.2 Algorithms used in the System 26

Chapter 04: RESULTS 28

4.1 DISCUSSION AND RESULTS 28

4.2 EVALUATION 34

Chapter 05: CONCLUSION 35

5.1 CONCLUSION 35

5.2 APPLICATION OF MAJOR PROJECT 36

5.3 LIMITATIONS 38

5.4 FUTURE WORK 40

REFERENCES 41

LIST OF FIGURES

Figure No.	Description
1.1	Captioning of images using image captioning system
1.2	Libraries used in the project
2.1	Boy eating ice cream
2.2	Semantic Keyword Extraction Module
2.3	Image Captioning System Architecture as in the paper
2.4	Project Architecture of the proposed CNN-LSTM model
2.5	Block diagram of the model using CNN, YOLOv4 and GRU
2.6	System Architecture of the proposed system
3.1	Dog playing with ball, Image data from Flickr8k dataset
3.2	Five human readable captions for image
3.3	Flowchart of the project
4.1	Data is read
4.2	Sample training data, Images in Flick8k dataset with caption
4.3	Caption processing done with the tokens in the front and end
4.4	Caption model
4.5	The summary of the proposed model
4.6	The training of our system
4.7	Model loss
4.8	The result of the proposed model

LIST OF ABBREVIATIONS

ANN	Artificial Neural Networks
BLEU	Bilingual Evaluation Understudy
CIDER	Consensus-based Image Description Evaluation
CNN	Convolutional Neural Network
DLLIB	Distributed Deep Learning Library
GRU	Gated Recurrent Units
LSTM	Long Short Term Memory
Numpy	Numerical Python
Pandas	Panel Data
OS	Operating System
RNN	Recurrent Neural Network
SDLC	Software Development Life Cycle
YOLOv4	You Only Look Once version 4

ABSTRACT

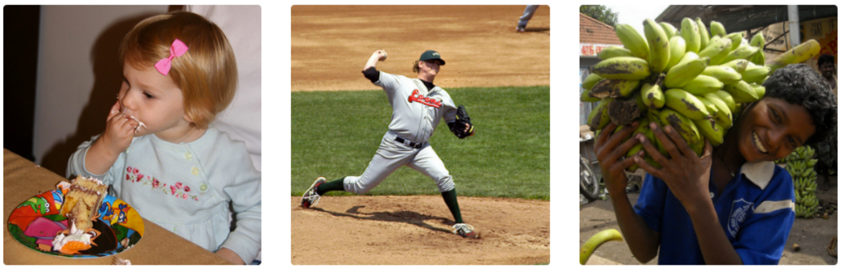
In this project we try to create and examine the working of an image caption generator made using CNN and LSTM .Image Captioning refers to the process of generating a human-readable textual description or a sentence of the image that is taken as input explaining the content of the same.Image Captioning has been a popular topic in this age of technology due to its various advantages like helping the blind or visually impaired people easily access and understand the image they are viewing on the internet. There are various scenarios experienced by software developers where an image and the capabilities of vision is not sufficient alone to build more interactive, intelligent and accessible software through images. Extra content and clarification or an alternative text is needed in these situations to provide a more accessible experience. Since currently over the internet a great number of images remain to be described it is impossible to be done manually, Thus taking help of deep learning, image processing and natural language processing we can give the power to a computer to describe images on its own. In this proposed model we create a two staged model using Deep Neural algorithms(Convolutional Neural Networks and Lost Short Term Memory).

An image is given as input to the model where mainly three functions are performed. In this model we use the Flicker8k Dataset .First the features of the image are extracted, then the model is taught how to differentiate those features and categorize them in a way that could predict the correct output, and finally we have the function where our model is able to generate the words step by step to define the image and all its features. The project focuses on making an image captioning model using cnn and LSTM. A pre-trained CNN is used to extract features vectors from the image while the LSTM based language model generates the captions by producing one word at a time.

Chapter 01: INTRODUCTION

1.1 INTRODUCTION

Image Captioning refers to the process of generating/creating a textual description for given images, it describes a scenario captured in the image. It is the task of writing text descriptions of what appears in an image, making use of the deep learning domain.it identifies objects in a picture and carries out a few processes to accurately to define the image or sort out the most crucial parts of the image needed by the user.Image caption generation is a task under artificial intelligence where the model is trained in a way that it can understand and define an image on a same level as human beings do. Image captioning has a lot of applications, as of now NVIDIA is using this technology to create an application to help people who have low or no eyesight. Image Captioning is used by large corporations to enhance the search experience by enabling image searches,Social media sites like Twitter, Facebook etc. uses this technology to suggest similar images according to the users preference and adjust the user's feed. Image captioning is used to make web surfing much more accessible and easier to give related image results. More applications include altering applications, virtual assistant, and faster image retrieval and indexing. Image captioning also has applications in the field of biomedicine, commerce and military. Deep Learning plays an important role in this model, it is preferred over machine and neural network learning algorithms due to its better performance and high speed and accuracy. Deep neural networks help in identifying objects, they are able to learn from the already trained instances and predict the outcome using that knowledge.



"little girl is eating piece of cake."

"baseball player is throwing ball in game."

"woman is holding bunch of bananas."

Figure 1.1 Captioning of Images using Image Captioning System[1]

Our proposed model is fed an image which is the input and according to the training done and the intelligence of the model it generates a simple easy caption that explains the image, its context and features similar to how a human would describe the image, the caption is completely human readable.In this model we aim to precisely determine the objects and their relationship. We use an advanced technique of CNN and LSTM which is a two staged model in which we have the first stage which uses convolutional algorithms and has the already pre-processed image as the input, this stage is also called the encoder stage.Several convolutional layers are used at this stage to extract the features from the picture vector before moving on to the next.Convolutional neural networks are a unique class of deep neural networks that are capable of processing data with input shapes resembling 2D matrices, as well as any pictures that have undergone rotation, translation, scaling, and perspective modifications. The second stage, also known as the decoder stage, is where we construct captions linearly using the LSTM approach.LSTM is a kind of RNN that is used to solve problems involving sequence prediction.Basically with the knowledge of the previous text we can predict what the next word can be. LSTM carries the important and relevant information throughout the processing of inputs and discards the non relevant information, which makes the captioning precise and accurate.

1.2 PROBLEM STATEMENT

In this age of technology everyone wants an accurate and efficient result to what they are searching for on the internet, images of what they are looking for proves to be a better answer. Looking for images based on the similar captions associated with it makes searching more interactive and easier. The captions associated with the images provide a more accessible experience. Image captioning is the popular topic that provides this experience of making your experience more interactive. This is the same technology used to show images of your interest on social media sites on your feed. It has various advantages like helping the blind or visually impaired people easily access and understand the image they are viewing on the internet. For many images on the internet an extra clarification or an alternative text is needed to make it more easier to understand and point out important features of the image the creator wants the focus to be, as this is nearly impossible to be done manually for the neverending images on the internet we want to create an Image caption generator model, that is training our computer to carry out this work.

1.3 OBJECTIVES

The objective of the project is to create a model that accurately describes an image with a caption that is easy to understand and describes our image . In this project we work on the popular and important technology of Image captioning using deep learning. As the name suggests our goal is to design a model that generates accurate captions of the image that is taken as input using Convolutional Neural Network and Long Short Term Memory algorithm, this model can be implemented using RNN but more precise results can be obtained using LSTM. Our task is to get that precise caption which is human readable and easier to understand and figure out the features of the input image after we have trained our model. We use the Flickr8k Dataset to train our model.

1.4 **METHODOLOGY**

The project's methodology shows the systematic plan in making of an Image captioning system, it involves the dataset made up of around 8000 images which contains several different scenes and instances but does not involve any popular people or location, long with the dataset there is a text file containing five captions associated with each picture. The methodology of the project helps us understand the various steps, approaches that are taken along with the various tools and techniques involved, some of which are:

1. Planning : A basic framework for the project is made in which the appropriate steps and complete understanding of what path to take to create the system is worked upon.
2. Researching : Various researches and the methods used are studied, the different approaches by researchers in their proposed idea, different articles,journals and published papers are read.
3. Algorithms and Libraries used : The deep learning algorithms mainly used in this project are CNN and LSTM, which is implemented using python 3.6, several libraries like pandas, numpy , textwrap, string and os are used.
4. Pre-Processing data : As machine systems do not take in image as a form of input, images are first pre-processed and turned into a 2D matrix which can be passed to the image based module, the noise is cleaned from the image to make the output more accurate.
5. Image based module : In this module we use CNN and its layers, that is, Convolutional layers and pooling layers along with using ReLU, the main objective of this module is to take the features from the images and create a feature vector to forward it to the next module.
6. Language based module : This module uses LSTM to generate words in a linear sequence to create a sentence keeping in check the previous word to create a meaningful sentence.
7. Caption Generation : At the end we receive the output caption of the image, keeping in check the color , attributes and relationships between the objects that are taken into consideration.

1.5 ORGANIZATION

This project report is divided into five chapters which are as follows :-

Chapter 1: - In this chapter we give a brief introduction of the project. The chapter provides the introduction of the project and gives a brief overview of the Image Captioning. This chapter describes the motivation, the problem statement, objectives and methodology of the technology that is worked upon. It walks us through the basic framework of the project and the desired goal.

Chapter 2: - In this chapter we go through the work done previously by other researchers on the topic Image Captioning and their published works. In this section we have mentioned various Journals and related papers which give information about the work that has been done earlier. The chapter gives information about how different researchers have used this technology for different purposes and ways and how they have tried to use the various models to train their computer to make a precise Image Caption Generator. The techniques and the results for those techniques are mentioned in this chapter and these help us to find the approach that we are going to use to create our model.

Chapter 3: - This chapter gives information about the steps that we are going to follow to build the whole project. It describes the steps that we are going to take to build the Image captioning model.In this chapter we talk in detail about the system and model development. We go through details about the dataset and the libraries used to work on the dataset. It also gives complete knowledge about the machine learning and security concepts we are going to use.It gives us the complete knowledge behind the various algorithms. The chapter also includes information about user validation and access provision, and the system required to run the project

Chapter 4: - This chapter gives the information about how the whole project work is done and about the checks kept at every stage. All the work done and the results obtained using different modules and libraries at different levels of the project is also shown in this chapter. It also consists of the results from the various performance measures that we have used in this project. The whole chapter is providing us with information about the performance of our whole project.

Chapter 5: - In this chapter we see the conclusion of the work done in this project presented in the project report. It provides information about the whole phases of the project and it also mentions the future scope for the project. All the information and results of all the phases is presented in this chapter along with the future scope for this project. This chapter also includes the applications of the project in the different sectors of the industry. Finally the prospective growth, future and improvements of the project are discussed.

To achieve the goal of creating human readable captions with high accuracies and efficiency , the requirements to make the project work have been mentioned below:

1.5.1 Python

Python was developed by Guido Van Rossum around 1985-1990, it is an interactive, garbage-collected and dynamically typed high-level language that has many features. Python is used for various jobs in various sectors like data analysis, automation,web development,many desktop applications etc. The major features of python include its property of Readability, python is user-friendly and easier to code, the ease of coding supported by the functionality of making intelligent models make python extremely popular, Versatility, It is a versatile language and can be used in various applications, sectors and tasks, and Supports various libraries , As python is an open source, the community and developers have contributed in creating several libraries that increases the potential of Python. Several such libraries have been used in our project like Numpy, Pandas, OS,Seaborn, TensorFlow, matplotlib, and warnings have been used .

1.5.2 NumPy “Numerical Python”

Numpy is the package which is used for scientific computing in python, created by Travis Oliphant in 2005. It is a fast and versatile library which helps us solve several problems like comprehensive mathematical functions, Linear algebra , random number generation and more. It is an open source python library and is maintained on GitHub, it is mainly used for working with arrays, since images cannot be taken as the input for the system, the images are first converted to array where the numpy library comes in use. It handles large datasets efficiently and supports multi-dimensional arrays and matrices.

1.5.3 Pandas (Python data analysis package)

Pandas is a fast, flexible, open source, powerful manipulation tool used for working with data sets in python, developed by Wes McKinney in 2008, it helps in analyzing, exploring , manipulating , exploring and cleaning data according to the user's wishes.
It allows us to analyze big data and can help make decisions based on statistics. It provides two classes for storing and manipulating data ,that is , the data frame and the series.
Pandas help is sorting the dataset, cleaning the messy data efficiently and making it readable and relevant which is absolutely necessary when working with data.

1.5.4 OS (Operating System library in python)

The OS library is the python library that includes functionality dependent on the operating system, it comes under Python's standard utility modules. This library in python provides an easy way to interact with the file system. If any file needs to be read, written or opened to see it can be done by using the os.path module. OS module is mainly used to create or remove a directory, change or identify the current directory as well as getting the content from the directory. The library is used to query information regarding the system environment which includes the operating system, disk space, path taken to execute the python program and memory information. OS also helps in managing external processes along with starting of these processes using the tools included in this library.

1.5.5 Tensorflow

Tensorflow is a python library developed by the Google Brain team in the year 2015, it is an open source technology, mainly used to train the deep neural networks. It is one of the popular libraries used in python due to its fast numerical computing. This library simplifies the processes built on it and creates deep learning models directly. It has a flexible architecture which allows easy processing of tough computation using data flow graphs having nodes and edges.

1.5.6 Warnings

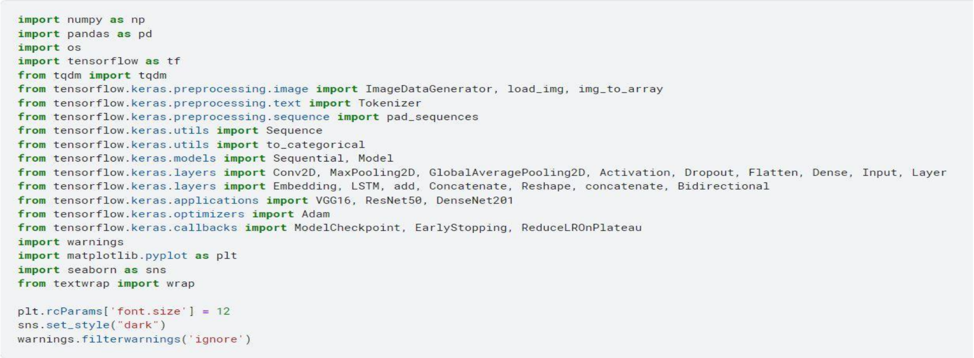
Warnings model in python is used to issue warning messages when it is to alert of some problem or condition in a program to the user where that condition does not need to raise an exception and terminate the program. Instead a message can be issued according to the rules and ignoring them can turn them into exceptions thus harming the program.

1.5.7 Matplotlib

Matplotlib is a library mainly used for visualization with python, it is used for generating animated and interactive visualizations. It can create different types of plots, figures that can zoom,pan and update, we can customize it to our type. It is used to visualize the 2D plots of array, it is built on numpy arrays and helps us get visual access to huge amounts of data. It consists of several plots like line, bar, scatter,etc.

1.5.8 Seaborn

Seaborn is a library based on matplotlib and integrates closely with pandas for visualization of statistical graphs in an attractive way. It helps you explore and understand your data and perform the semantic mapping along with statistical aggregation to create informative plots.



```python
import numpy as np
import pandas as pd
import os
import tensorflow as tf
from tqdm import tqdm
from tensorflow.keras.preprocessing.image import ImageDataGenerator, load_img, img_to_array
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras.utils import Sequence
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.models import Sequential, Model
from tensorflow.keras.layers import Conv2D, MaxPooling2D, GlobalAveragePooling2D, Activation, Dropout, Flatten, Dense, Input, Layer
from tensorflow.keras.layers import Embedding, LSTM, add, Concatenate, Reshape, concatenate, Bidirectional
from tensorflow.keras.applications import VGG16, ResNet50, DenseNet201
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.callbacks import ModelCheckpoint, EarlyStopping, ReduceLROnPlateau
import warnings
import matplotlib.pyplot as plt
import seaborn as sns
from textwrap import wrap

plt.rcParams['font.size'] = 12
sns.set_style("dark")
warnings.filterwarnings('ignore')
```

Figure 1.2. Libraries used in the project.

Chapter 02: LITERATURE SURVEY

2.1 FEASIBILITY STUDY

Feasibility research refers to the research and the information collected before the main study, it refers to all the articles and papers and knowledge collected to finalize the project and its framework. For our proposed model "Image captioning using CNN and LSTM" first we study about image captioning and deep learning algorithms. First we ask ourselves what exactly is image captioning, in simpler words, if we have to define image captioning it can be defined as the process of generating a textual description of any image. For example, if we search for an image on internet by using , 'boy eating ice cream', the internet would provide us all the image results possible related to the simple text description, in which all the images will have this tagline associated with it which contains the text "boy, eating, ice-cream" for every image having similar words in this description it will be given as a result for the searched text. Countless images with the similar description to what you are looking for are easily available with one click if on your devices due to these little captions that define an image. This is what the entirety and importance of Image captioning is, it covers more advantages and uses which will be discussed later in this paper. These countless images cannot have captions generated to it manually so the importance of a system that can do this work and give results similar to what a human would perceive from its vision. In Figure 2.1, the resultant image is internet output to when the text "boy eating ice cream" is searched.

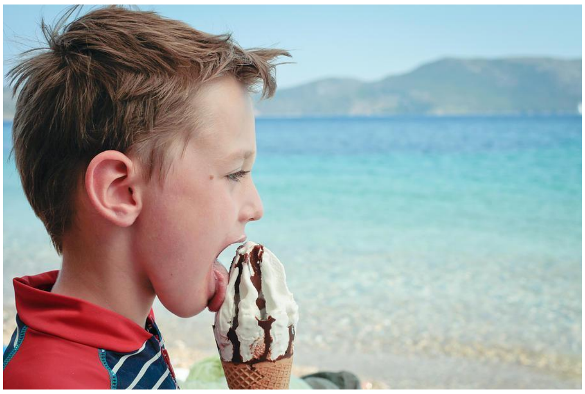


Figure 2.1 Boy eating ice cream

Now in this project , we aspire to create a system that would caption this image more in depth according to what we can see, which in this case includes , the sea, an ice cream , a boy in a red t-shirt. So this image can be accurately captioned as either focusing on the major objects and their relations in the image or all objects present in the image. The captions that one may create for this image can include information like:

- A boy eating ice cream
- A boy in red t-shirt eating ice cream
- A boy eating ice cream in front of the sea, and so on.

A human can interpret this image in many ways and can give the description accordingly so what we aim for in this model is to create such captions of certain length that will define such images similar to a human being.

After learning the basic concept of our project we shift to the various ways we can use to create this model where the deep learning models come in. Image captioning can be easily achieved using the algorithms of deep learning and computer vision.Deep learning is a part of machine learning, it is basically a neural network with several layers. What these neural networks do is similar to the human brain , that is, it tries to learn from the data that is given to it. In this project deep learning algorithms are used to train our model to learn from the input of images and caption dataset provided to it and when a new random image input similar to what the model has been trained into is put in the model then the model provides an appropriate caption. CNN and LSTM are part of this technology that helps in creating this system. This proposed idea has already taken deep roots in our everyday internet life such as on social media to suggest images for your feed and has a bigger cause to its creation , mainly helping the visually challenged people to access the internet more easily. These are the papers read to understand more about “Image captioning” and implementing it using CNN and LSTM.

2.2 **LITERATURE SURVEY**

1. **Abisha Anto Ignatious.L, Jeevitha.S, Madhurambigai.M, Hemalatha.M [2]**

The authors of this research suggest a CNN-LSTM architecture that is semantically driven and has features extraction, semantic keyword extraction, face recognition, and an encoder-decoder LSTM network. A semantic keyword extraction module is utilized to identify the items in the image, and a pre-trained CNN is used to extract features from the input image. The semantic tags found in the image are used to label the objects that are visible in it. The captions that are generated to describe the image are more effective with this technique. The language model built on LSTM then creates captions one word at a time. Facial recognition plays an important part in this project to recognize and identify the popular personalities from the input image dataset, in this model the faces dataset has 232 images. The precision of the generated captions is calculated using BLEU scores. Datasets used were Flickr30k and Faces dataset. According to the system architecture, the suggested system uses facial recognition, semantic feature extraction, and deep learning to create an output description of the input image that has been provided. An encoder-decoder architecture that limits the length of the output description generates sentences of a set length for each provided or tested image in the output. Two modules make up the picture captioning process: one extracts the image's features, and the other converts the features and objects provided by the first model into a meaningful sentence.Semantic keyword extraction from the description dataset helps to improve the content quality of the captions because the generated caption does not include information about some attributes in the image.The keywords are fed into a multi-layer perceptron network, which has hidden layers like the dense layer, dropout layer, and soft-max layer. These layers aid in identifying the attributes in the image and produce the best captions with additional details.The semantic labels in every image are the ones that are recognised as significant things shown in the image. The semantic feature extraction model is fed these labels, which are the fundamental semantic labels. The suggested study discusses various face recognition methods for identifying people in images and detecting faces. Using the DLLIB and facial recognition python packages, the final model has an accuracy of 99.38% on the labeled faces in this wild benchmark. Since RNN networks are appropriate for text processing and generation operations in deep neural networks, RNN-LSTM networks were used to build the language model. The language model with recurrent neural layers received the retrieved features and a semantic feature vector as input. RNN layers possess a memory component that determines which word should follow the first word and produces a meaningful sequence of words.As the captions generated are not personalized the model does not include the name of the celebrities in the image, for which the face recognition process is needed making use of the replacement methodology to use the name labels wherever the formal pronoun is present. For images having more than one celebrity there is a use of commas to generate the final caption. The proposed system is created using python, keras along with tensorflow frameworks and shows an accuracy of 73.95% . In the conclusion of the paper the authors talk about the attention image captioning has gained in the artificial intelligence community and how many technologies are based on it, the importance of RNN in the proposed system is highlighted along with the accuracy achieved using the semantic extraction process.

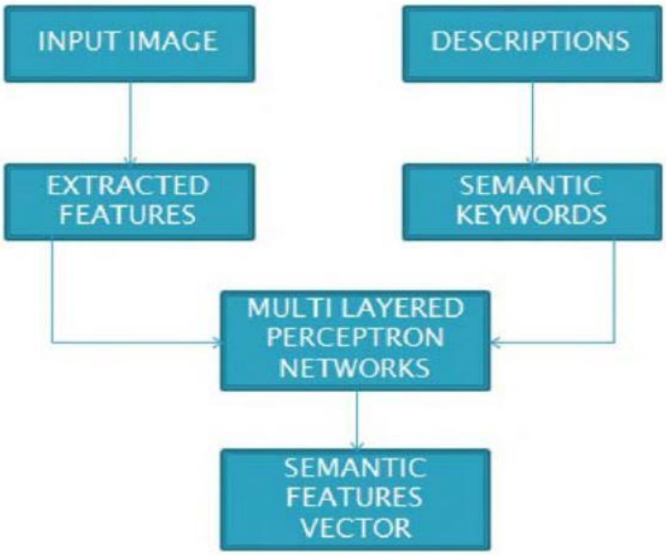


Figure 2.2. Semantic Keywords Extraction Module.[2]

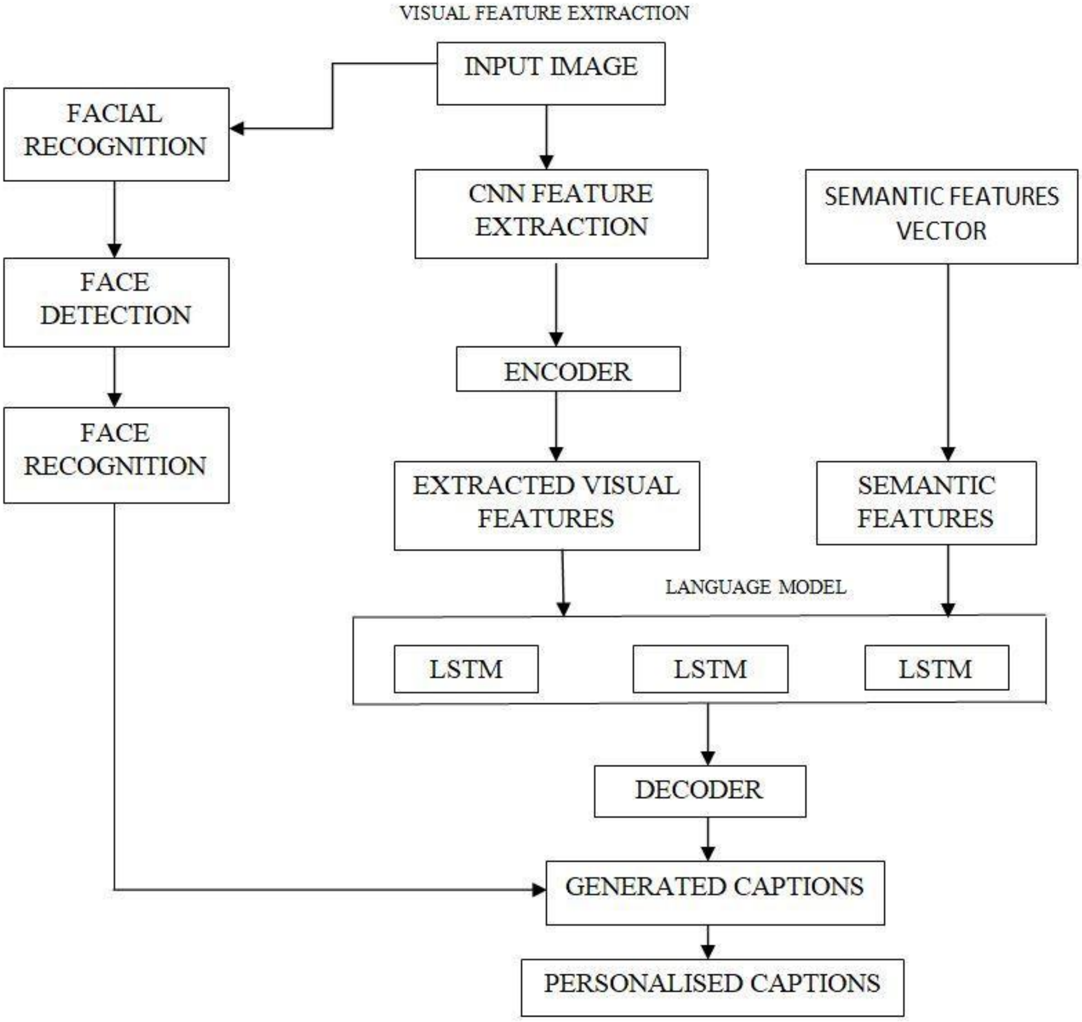


Figure 2.3. Image Captioning System Architecture as proposed in the paper.[2]

2. **Anish Banda,Harshvardhan Manne, Rohan Garakurthi [3]**

A deep neural network-based picture caption generating method is looked at in the model given in this paper. The technique takes a picture as input and outputs it in three different ways: phrases describing the image in three different languages, an mp3 audio file, and an image file.This methodology makes use of both natural language processing and computer vision. The goal is to create a model that can generate captions using LSTM and CNN approaches. In order to extract features from photos and produce a good description using the learned data, the target image is compared with the training images from the huge dataset using CNN, an encoder. To decode the description of the image that is generated, LSTM is employed as a decoder. The BLEU metric algorithm, which rates the caption's quality, determines the accuracy of the generated caption. The classification of images from the CIFAR10 dataset is done individually in the proposed model at first using a variety of extractors. The model was initially trained using the KNN approach, and subsequently well-known linear classifiers were used. As the high loss factor at the moment of classification grows during caption creation, the observed model's accuracy decreases. The model is then developed using a straightforward CNN, and it performs admirably during training. Both the models and the current ones were compared using the BLEU assessment score. CNN serves as an encoder in the proposed system to convert images into vectors. The most recent and sophisticated classification algorithm can be utilised in addition to the VGG16 design, although doing so would significantly lengthen training time. The output of the LSTM networks, whose model is comparable to the model used in machine translation, is the image encodings. The input is a 224*224-pixel image that has been compressed. The model is developed using the Flickr8k dataset, and it generates a caption based on the word structure and references found in the caption in the preparation data. The generated caption is then compared to the original description using the BLEU metric. RNN receives the output of CNN and uses it to generate language. However, RNN struggles with long-term memory retention, thus LSTM networks are employed instead. The conclusion highlights the field's tremendous growth and how much more has to be done in that regard. It puts out the concept of understanding how isolated photographs can be used as unsupervised data to improve the image explanation approach.

3. M.Pranay Kumar, V.Snigdha, R.Nandini , Dr. B.Indira Reddy [4]

Making a description for a picture is described as the process of photograph captioning. It describes a comprehension of the properties and connections between an image's constituent parts. The CNN deep learning algorithm and LSTM are employed in this Python-based study. In order to employ computer vision to assist the computer in recognising the context of a picture and generating an appropriate caption, many forms of RNNs are being developed. It is crucial to train the model as exactly and precisely as possible because a description that sounds more like a human creates a better first impression and is simpler to grasp.Any pixel can be used for color or monochrome images, and image captioning can provide acceptable sentences to explain them. To comprehend the context of the image and provide captions in a natural language like English, this task requires computer vision and natural language processing techniques. CNN, which uses a 2D matrix as its input image, is used to create the caption generator, and Xception, a CNN model trained on the Flickr8k dataset, is used to extract the image's features. The LSTM model that generated the captions is then fed the features. The Flickr8k dataset, which consists of 8000 photographs and each of which has five captions to explain various aspects of the image, is used in the proposed method. Prior to combining them to identify the images, CNN scans the input images from top to bottom and left to right to extract key components. Based on the previous paragraph, LSTM assists in predicting what the following word would be. It performs appropriate statistics while processing inputs and has an overlook gate that allows it to reject statistics that are not relevant. The conclusion brings the paper to a close after examining the results and achieving its primary goal of demonstrating various image annotation approaches. Each technique's advantages and disadvantages are discussed, along with a number of experimental results. Image captioning will continue to be a hot issue and expand alongside social media platforms for a very long time thanks to the advent of new deep learning network architectures.

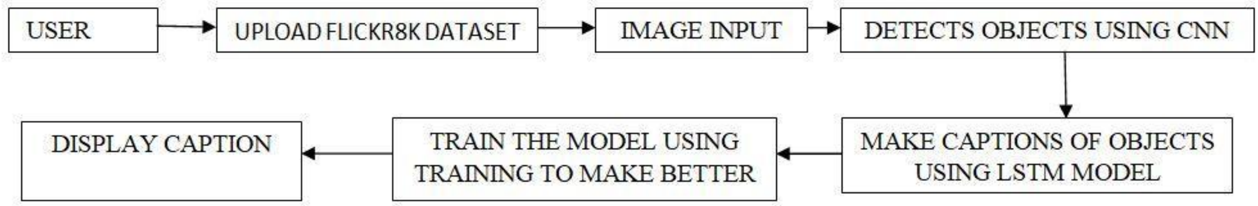

USER
UPLOAD FLICKR8K DATASET
IMAGE INPUT
DETECTS OBJECTS USING CNN
MAKE CAPTIONS OF OBJECTS USING LSTM MODEL
TRAIN THE MODEL USING TRAINING TO MAKE BETTER
DISPLAY CAPTION

Figure 2.4. Project Architecture of the proposed CNN-LSTM model.[4]

4. Muhammad Abdelhadie Al-Malla, Assef Jafar, Nada Ghneim [5]

A component of computer vision and natural language processing, image captioning's job is to use a single world to describe the object in an image.The majority of captioning research focuses on deep learning methods, particularly encoder-decoder models incorporating CNN feature extraction. However, only a small number of works make use of object detection tools to improve the captions that are automatically created. Convolutional features from a CNN model that was trained on Xception and object features from a YOLOV4 model that was trained on MS COCO are used to create an attention-based, encoder-decoder deep architecture in this research.Additionally, it introduces the significance factor, a brand-new object feature encoding approach. The model was evaluated using the MS COCO and Flickr30k datasets, and when its performance was compared to that of other models, it was discovered that it raised the CIDEr score by 15.04%. The necessity for labeling and annotating images has grown significantly because they are one of the most readily available and accessible types of data on the internet. The volume element of large data is the main emphasis of image captioning systems, which calls for effective resource management and cautious experiment design. Encoder-decoder models, which use LSTM, GRU, or one of its variants, are among the most effective techniques. However, in recent years, certain works employing YOLOv3, YOLOv4, and YOLO9000 have also gained popularity because of their speed, accuracy, and suitability for real-time applications. Each object tag in an object feature comprises information on the bounding box, the object class, and the confidence level. This study is centered on the claim that using these traits improves accuracy and faithfully imitates how people see scenes, and it assesses the outcomes. The research methodology entails removing object features from the YOLO model and adding them to an encoder-decoder deep learning model coupled with CNN features.Object features are added in a simple concatenation manner and the result is a good improvement. The impact of sorting the object tags that are taken from YOLO according to a metric is also taken into consideration. MS COCO and Flickr30k are the two datasets used in the model's implementation. Both of these datasets contain actual photographs with five handwritten annotations each that were taken from the Flickr photo-sharing service. To measure the consistency between a human and machine translation, BLEU measures are utilized as evaluation metrics. Raw object layout information is used, as opposed to other proposed systems where the CNN features and embedded object features are combined. An attention module, a GRU, and two fully connected layers are used to generate language. The model generates captions using attention and is easy to use, analyze, and train. The final layer before the fully linked layer has its spatial information obtained using an Xception CNN that has been pre-trained on ImageNet. It aids the model's understanding of the image's objects and their relationships. YOLOv4 is utilized in the object detection model due to its quick processing and high accuracy, making it appropriate for real-time and huge data applications. In order to utilize object detection and picture classification characteristics, the output of the YOLOv4 subsystem is attached as the final row in the output of stage 1 using a concatenation step.By mapping the feature space to a smaller space for the language decoder, embedding ensures that the size of the features stays constant. The model is smoothed and differentiable using the Bahdanau soft attention approach.The term "attention" refers to a technique that mimics cognitive attention by emphasizing the most crucial information from the input data and fading the less crucial information. Due to the encoder-decoder system's adherence to the human instinct of emphasizing various aspects of an image when describing it, performance has increased. Because of its fast and low memory utilization characteristics, GRU is utilized for decoding.This model, which was trained using a backpropagation algorithm, creates a caption by creating one word at each time step in accordance with the context vector, previous hidden state, and previously created words. The suggested system's code is written in Python utilizing the Tensorflow, Keras, and YOLOv4 model, which was trained on MS COCO and imported from the YOLOv4 package. The recommended solution, which is a speedier, more human-like, and more effective variation of the standard method, is explained in the conclusion as to why it is effective.

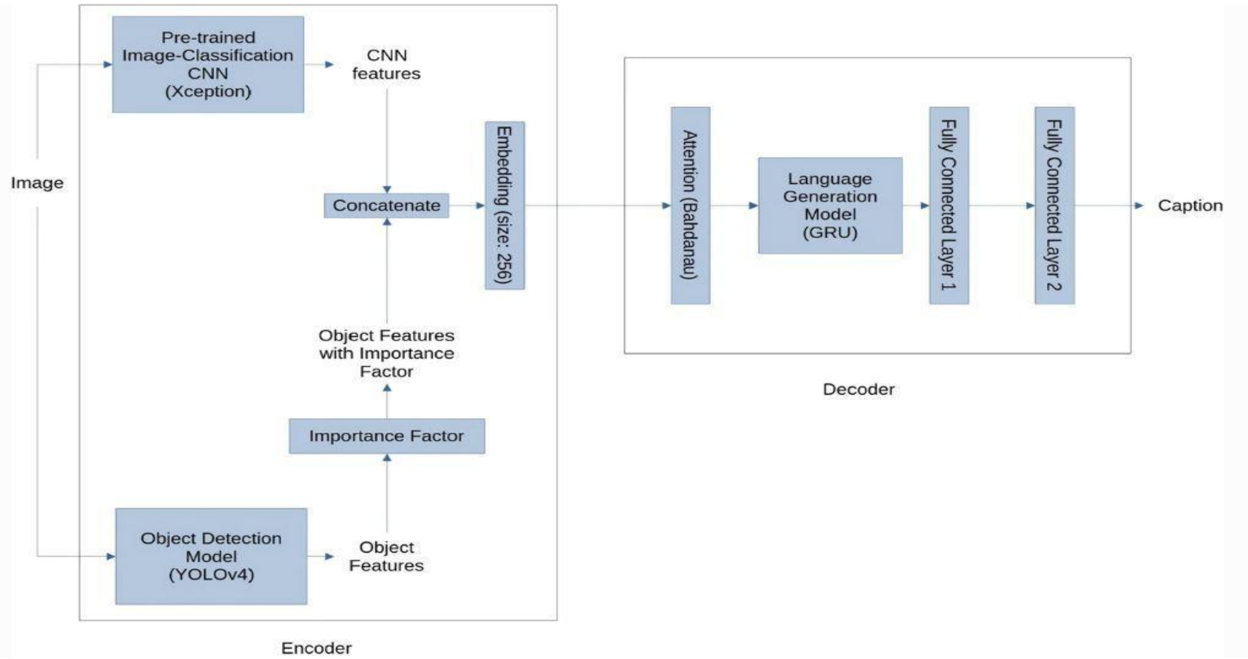


Figure 2.5. Block diagram of the model using CNN , YOLOv4 and GRU.[5]

5. **Priyanka Raut, Rushalo A Deshmukh [6}**

It's an exciting challenge to teach machines to comprehend the substance of images and provide captions that are close to human-level intelligence. The suggested approach in this paper uses CNN and LSTM to provide accurate captions. Typically, models divide an image into components and categorize these elements before producing a caption, which is primarily focused on producing simple, natural language captions that accurately reflect the image content. The conventional approach makes use of CNN and RNN, but it has a number of drawbacks, including gradient vanishing, inaccurate object and relationship identification, and production of captions only for viewed images. A different approach instead of the traditional method for captioning uses the combination of CNN and LSTM. It is designed to address the issues that can occur with the conventional method. There are two steps to the Model: Long

Short-Term Memory is used in the second step after the Convolutional algorithm in the first stage.Image or photo input is used in the first stage.Additionally, the suggested system model places emphasis on the useful captions that best describe the visual scenario. The proposed system architecture's initial stage, referred to as the encoder stage, receives an image vector as input that has previously undergone preprocessing. The vector is then subjected to a number of convolutional layers in order to extract the required characteristics from it before proceeding to the next level.Following the application of a number of convolutional layers or operations, the image vector is passed on to the decoder step.Stage 2 linearly processes the image vector provided by Stage 1 to produce captions.The Stage 2 LSTM algorithm, an advanced recurrent neural network (RNN) technique, is used in the methodology to help combat the gradient explosion issue. The LSTM has an advantage since it contains a variety of memory gates that control how information flows to and from Stage 2. The ability to store data for extended periods of time and dependencies is an additional advantage. Stage 2 produces a short English statement or caption for the input image in a sequential manner.For the provided input image I, the suggested system's image captioning model outputs precise captions,C in a straightforward language, matching the level of human imagination and intuition. The captioning method makes use of sophisticated CNN and LSTM algorithms. The picture-based module, which receives the input image first, applies the Convolutional and Pooling layer of the CNN algorithm to build a vector known as the feature vector of the input image. A ReLu layer comes after each Convolutional layer. The size of the feature vector is then decreased using a pooling layer before being passed on to the following model. CNN's Fully Connected Network, the top layer, is not included in our model because we just require the feature vector. While Fully Connected Networks are utilized as Classifiers, Convolutional and Pooling layers are employed as Feature extractors.For the provided input image I, the suggested system's image captioning model outputs precise captions,C in a straightforward language, matching the level of human imagination and intuition. The captioning method makes use of sophisticated CNN and LSTM algorithms. The picture-based module, which receives the input image first, applies the Convolutional and Pooling layer of the CNN algorithm to build a vector known as the feature vector of the input image. A memory cell in the LSTM has a longer storage capacity for data. The two special tokens, startseq and endseq, are included in the series of sentences or captions so that the algorithm can determine when to begin and end the sequence of sentences. Last to be generated is the caption. The items, colors, activities, and relationships between the objects are the primary subjects of the captioning model.The input image is transformed into a fixed-sized pixel matrix with each pixel's color code placed in its appropriate location. Every single image is preprocessed, turned to grayscale, then divided into foreground and background using a threshold value. Every picture object is subject to edge detection. On the pre-processed input image, CNN serves as a feature extractor. The language-based model known as LSTM is used to translate the encoded features into plain language. The output from the LSTM is the input for the Caption generation module, which generates a caption for the provided image in a linear sequence. The Flickr8k data collection is utilized for model testing.The model's accuracy in identifying the items in the photographs and their relationships is demonstrated by the results and conclusion, which also show that the model has decreased the mistake rate in the caption. The proposed approach can be expanded in the future so that a computer programme can be trained on photos to generate descriptions for output captions that are more precise.

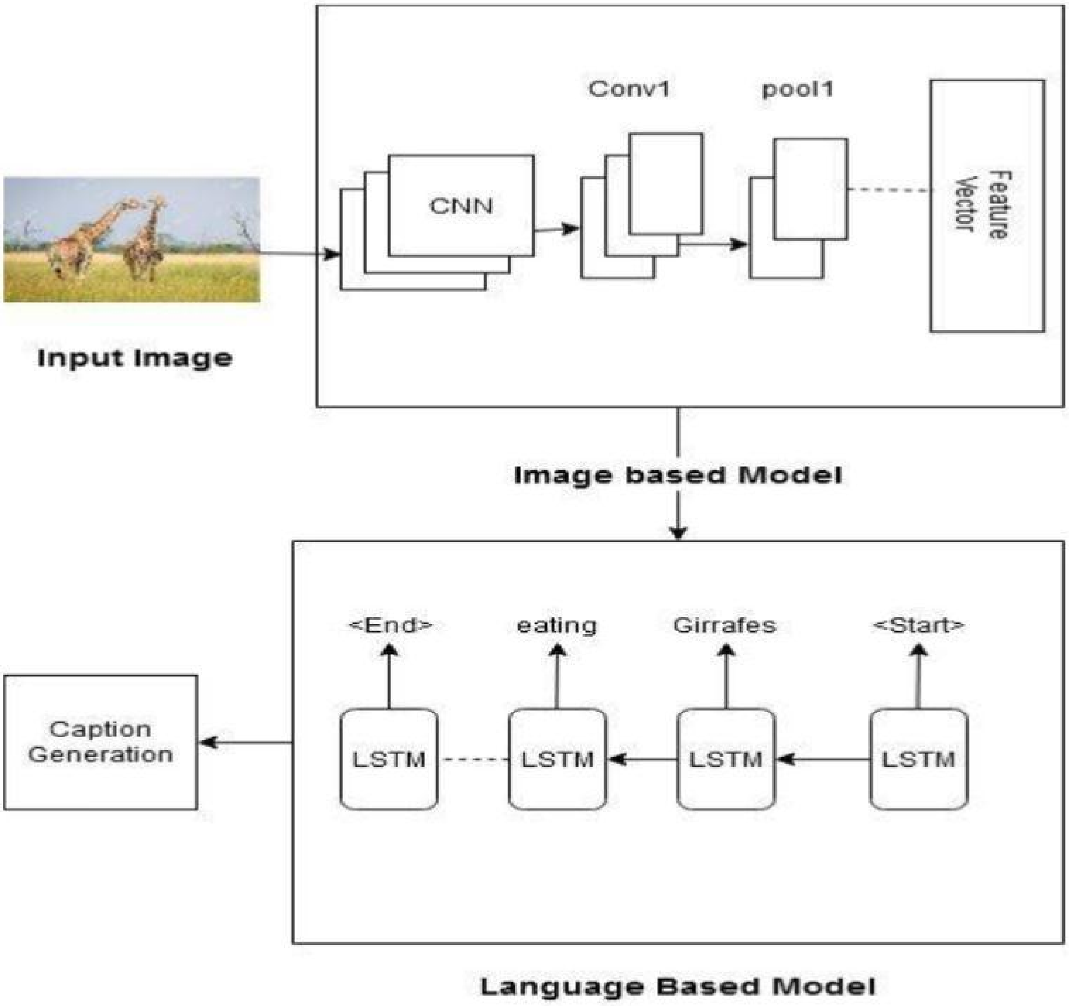


Figure 2.6. System Architecture of the proposed system.[6]

6. V.Varshith Reddy,Y.Shiva Krishna,U.Varun Kumar ,Shubhangi Mahule [7]

The project's objective is to give an image a caption. It is required to identify the important details, their traits, and the relationships between the items in an image. Thanks to the advancement of deep learning techniques, the availability of huge datasets, and the accessibility of powerful computers, we can now build models that can produce captions for images. This has been put into practise in a Python project where we combined CNN and LSTM deep learning algorithms such that a computer employing computer vision can understand the context of a picture and present it in a language like English.Both monochrome and coloured photographs can have subtitles thanks to grayscale image captioning.The project's objective is to give an image a caption. It is required to identify the important details, their traits, and the relationships between the items in an image. Thanks to the advancement of deep learning techniques, the availability of huge datasets, and the accessibility of powerful computers, we can now build models that can produce captions for images. In order for a computer using computer vision to comprehend the meaning of a picture and communicate it in a language like English, we merged CNN and LSTM deep learning algorithms in a Python project. Grayscale picture captioning makes it possible to add captions to both coloured and monochrome photographs. The LSTM model, which will provide the image descriptions, will be fed the image attributes from the CNN model Xception, which was trained on the Flickr8k dataset. Convolutional neural networks, a subset of deep neural networks, are capable of processing data that has a 2D matrix-like input shape.The generation of image captions is accelerated by deep neural networks. The proposed approach spares social media users the time-consuming task of scouring Google for descriptions that fit an image. Our solution offers social media users a simple platform on which to upload the image of their choosing. The caption for the submitted image doesn't need to be manually entered by the user. The proposed approach can address the problems with picture retrieval. can submit pictures in any size, both in colour and in black and white. By producing appropriate, expressive, and extremely fluid captions using tensor flow and algorithm, neural networks can solve all the problems. It is feasible to compute automatic metrics effectively. There is no need to waste time looking because the captions are generated automatically. In the conclusion of the paper different evaluation metrics, strengths and weaknesses of the model is discussed .It explains the process and potential research directions but still a method that can generate high quality images is yet to be achieved. The final words stress on the active topic and the growth and importance it will have in the coming years.

Chapter 03: SYSTEM DEVELOPMENT

3.1 ANALYSIS

Image Captioning system is a hot topic in the field of artificial intelligence and there are various techniques that have been implemented and developed by researchers all around the world.
Deep learning is a constantly growing and emerging field, if appropriate and expressive data is provided to the technology the output result can be much more accurate and efficient. For image captioning the proposed framework includes an encoder - decoder model making use of CNN and LSTM. This technique provides an easy to use system to caption images in a form that is human readable in a natural language like English. The goal of the system is to create a system that accurately and efficiently generates captions for the input image.
Different Techniques involved in making of Image captioning system and the system architecture are discussed here:-

3.2 DESIGN OF THE PROJECT

The CNN-LSTM model that is suggested in this paper makes use of deep learning methods. Figure 6 shows the Block Architecture for the picture captioning system.The output description is kept to a reasonable length using an encoder-decoder paradigm. There are four modules in the system for captioning images. The image-based module, which receives the input image first, uses the Convolutional and Pooling layer of the CNN algorithm to extract the objects from the images and the relationships among them, creating a vector known as the feature vector of the input image. Each Convolutional layer is followed by a ReLu layer. A pooling layer is then used to reduce the size of the feature vector before it is sent on to the following model. Convolutional and pooling layers serve as Feature extractors, whereas fully connected networks function as Classifiers.Due to LSTM's ability to store lengthy sequences of data, the encoded features vector from the previous model is now passed on to the Language Based Module, where it is decoded into a natural language caption. A memory cell in the LSTM has a longer storage capacity for data. The caption has a set maximum length so that the algorithm knows when to stop adding new sentences to the sequence. At last the desired caption is generated keeping in check the relationship, color and actions between the objects.

3.2.1 Dataset

For this project we make use of the Flicker8k Dataset for making of the model and the experiments. The Flicker8k data dataset is simple to download and ideal for desktop computers and laptops with modest workstations.

Flickr is an online community for high definition video or image hosting. It is one of the best online photo management and sharing applications in the world. The Flickr8k dataset is a dataset created from selecting images from six different Flickr groups,it does not have any celebrities or well-known personalities or locations , the images depict different scenes and situations.

The Flicker8k data collection allows for efficient model training.This dataset consists of 8092 elements in which we have 8091 images and 1 text file denoting the captions of the images. Contains in the Flicker8k_dataset.zip file areFlickr8k_Dataset: There are 8092 photos in this folder, each having a unique size, shape, and color. 8092 photos were used in total; 6000 were used for training, 1000 for development, and the other 1092 were used for testing the proposed model.

This dataset is 1.12GB in size and is open for all to access.

1. **Images:** The Image folder of the dataset consists of 8091 images of different scenarios and situations not including any celebrities or popular destination. The images are of variable size and consist of several similar scenarios For example Figure 3.1 is just a small example of similar scenario images chosen from the dataset of a “dog playing with a ball”.

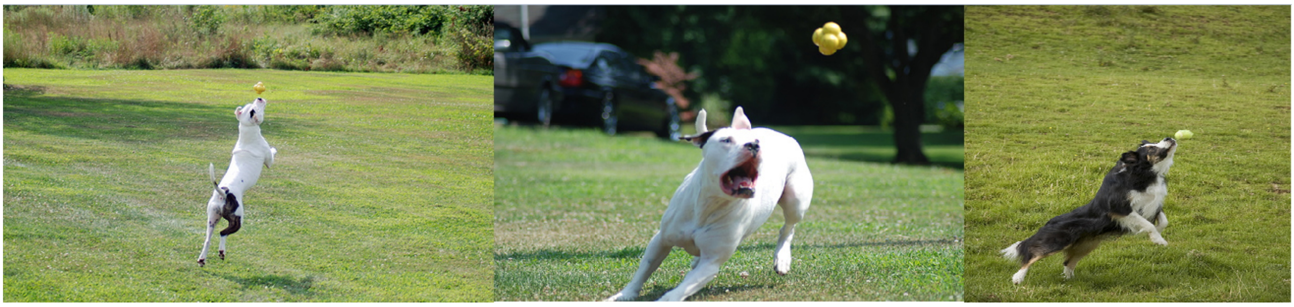
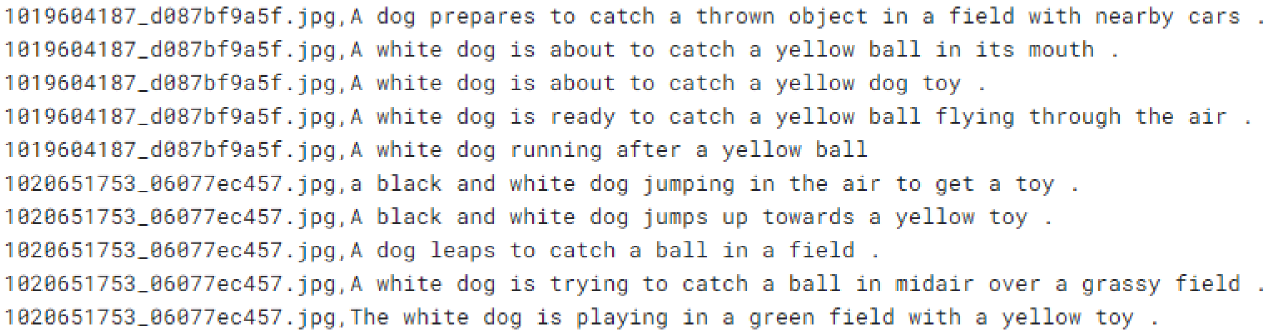


Figure 3.1 “Dog playing with a ball” image data from Flickr8k dataset

2. **Text File:** The caption text file consists of five separate captions for each image present in the dataset, the captions define the image differently keeping in check the various different objects present. The text file is of 3.32 MB and the captions are human readable and not too complex. Figure 3.2 shows the different types of five captions describing the image.



```
1019604187_d087bf9a5f.jpg,A dog prepares to catch a thrown object in a field with nearby cars .
1019604187_d087bf9a5f.jpg,A white dog is about to catch a yellow ball in its mouth .
1019604187_d087bf9a5f.jpg,A white dog is about to catch a yellow dog toy .
1019604187_d087bf9a5f.jpg,A white dog is ready to catch a yellow ball flying through the air .
1019604187_d087bf9a5f.jpg,A white dog running after a yellow ball
1020651753_06077ec457.jpg,a black and white dog jumping in the air to get a toy .
1020651753_06077ec457.jpg,A black and white dog jumps up towards a yellow toy .
1020651753_06077ec457.jpg,A dog leaps to catch a ball in a field .
1020651753_06077ec457.jpg,A white dog is trying to catch a ball in midair over a grassy field .
1020651753_06077ec457.jpg,The white dog is playing in a green field with a yellow toy .
```

Figure 3.2 Five short human readable captions for the image.

3.2.2 Flowchart of the Major Project

First, the dataset is imported which contains the images and captions of those images. The initial data-processing is done on the image data, then the pre-processed data is forwarded to the image based module which contains the CNN. Inside this module we make use of the convolutional layers and Pooling layers along with ReLU . In this module the features of the objects are extracted and a feature vector is created. This is then sent to the language based module containing LSTM, which creates a caption of a certain length. Finally a human readable caption is generated.

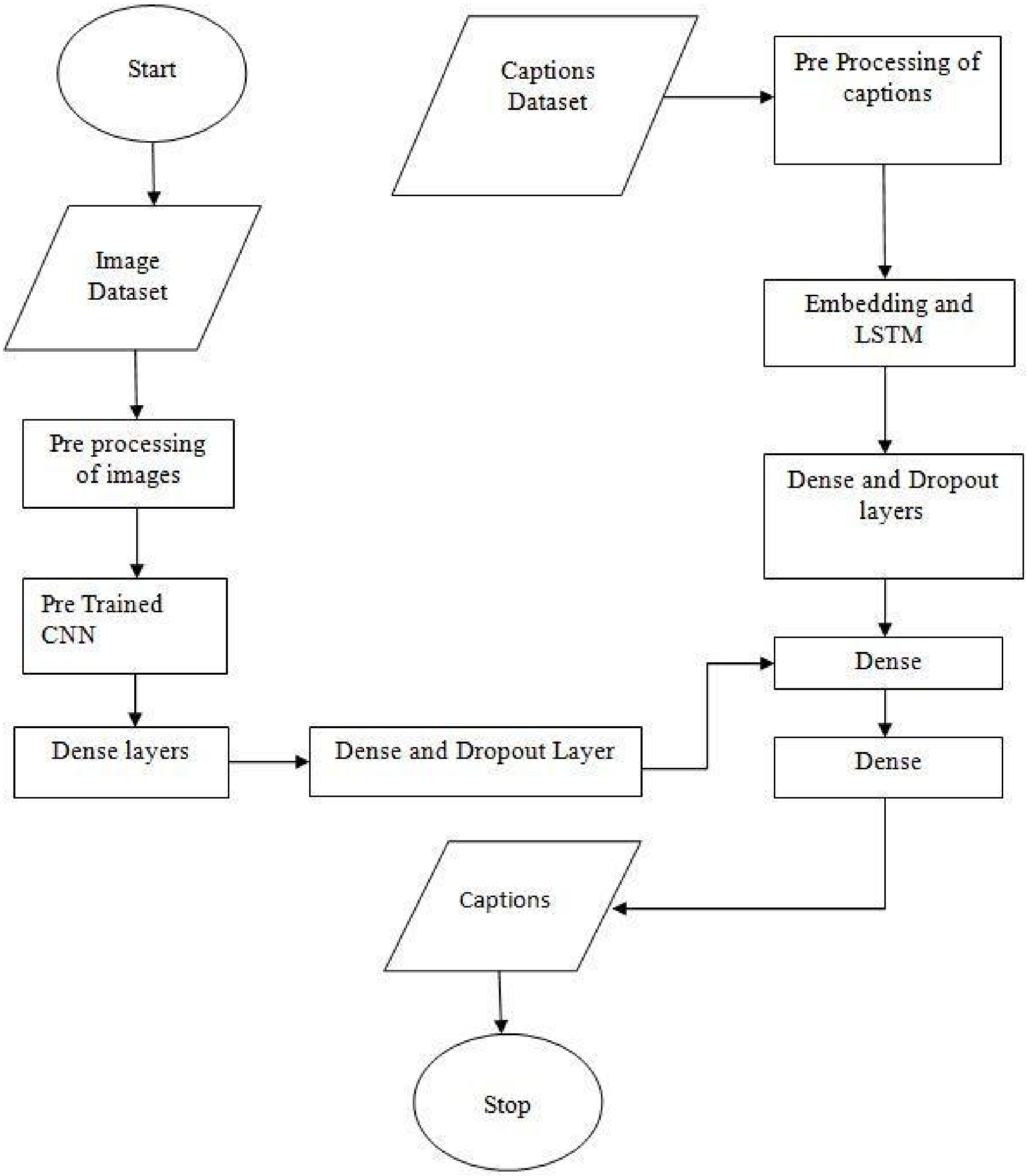


Figure 3.3: Flowchart of the project

3.3 TRAINING AND TESTING OF MODELS

This section of the report discusses the modules implemented in this project along with the algorithms used to make the modules .

3.3.1 Modules of the Proposed Model

The four modules of the image captioning system are:

1.Image Pre Processing

The input image is converted into a 224x224x3 fixed size pixel matrix, where each pixel's color code is located at a specific location because machines cannot understand images. The noise in each image is then eliminated during pre-processing. The output of the picture pre-processing model is the final pixel matrix, which is then entered into the following module.

2 Image Based Module (CNN)

Convolutional and pooling layers from CNN's modified version are employed as feature extractors when doing feature extraction. The result of the preceding pre-processing module, which is a pixel matrix, is the input for the image-based module.This module takes the Image Pixel matrix and extracts the features, saving them as feature vectors.The first layer in this module to extract features is the convolution layer. Each Convolutional Layer is followed by the ReLU layer.Without sacrificing any of the image features, a pooling layer is employed to minimize the size of the feature vector. This module's output is used as input for the following module.

3. Language Based Module (LSTM)

This module's primary goal is to use LSTM to translate the encoded features into a natural language like English that is human readable after receiving the output from the preceding module as input. The vanishing gradient issue is solved by LSTM, which also allows for the storage of lengthy data sequences without losing track of their order. The prior word and context can be utilised to infer the following word in the sentence's sequence. The labels and target text for this module's training are first predefined. The label keeps the information organized in a sequence that progresses until the sentence's needed length.

4. Caption Generation

This is the final module where the previous module's output serves as its input. This module's goal is to produce captions in a linear order using information from the preceding module. A straightforward human readable caption is produced at the end.

3.3.2 Algorithms used in the System

1. **Convolutional Neural Networks**

Deep learning neural networks like CNN are used to process organized arrays of data, such photographs. In computer vision, CNNs are frequently employed for a variety of visual tasks, including text classification, natural language processing, and picture and image classification. Applying a filter to an input to create an activation, which is then represented as a numerical value, is the simple process of convolution. By repeatedly applying the same filter to a photo, a feature map, often referred to as a map of activations, is produced. These places and intensities represent the discovered features.A linear operation known as convolution produces a two-dimensional array of weights known as a filter by multiplying a set of weights with the input. Applying the filter repeatedly to the entire input image will detect the feature wherever in the image if it is configured to look for a specific type of feature in the input.

The pre-processed pixel matrix serves as the system's input, and its output is an encoded feature vector.After applying a convolutional layer to extract the image's features, ReLU is then performed after each convolutional layer. A feature vector is then retrieved from the input image after using a pooling layer for each feature in the feature map.

1.1 Convolutional Layer

The convolutional layer is the core element of a convolutional neural network.It is the algorithm's initial layer of the CNN deep neural network. K learnable filters, or kernels, are used to create the parameters of this layer. Each kernel has a width and a height, and they are almost always square. Despite being small, these solitary filters screen the entire depth of the volume.The depth is the number of CNN input channels in the image. The depth for volumes further down the network will depend on how many filters were employed in the layer before. The input for the suggested system is the pre-processed input image's pixel matrix.By applying a chosen filter to the input image in this layer and calculating the dot product, a feature vector is produced. The values of all previously calculated filters are then entered into a map that has been built. These filters essentially create a new feature vector by extracting the features from the input image. In order to prevent the summation of all pixel values to zero, ReLU is then used to perform a nonlinear operation in which the negative values from the filtered images are substituted with zero.

1.2 Pooling Layer

The building components of convolutional neural networks include pooling layers.
CNN-discovered characteristics are combined via pooling layers. In order to reduce the number of parameters and calculations in the network, it aims to gradually lower the spatial dimension of the representation. It accomplishes this without degrading the features' quality. The highest value is chosen from each window and replaced in the map. It aids in accelerating network computation and resolving the over-fitting problem.

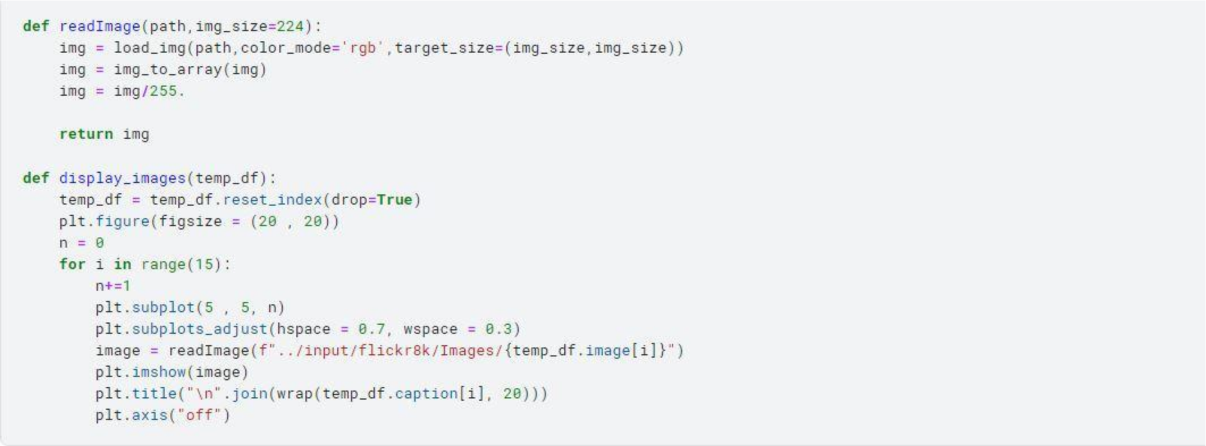
2. Long Short Term Memory

The gradient vanishing issue that RNN has is addressed by LSTM, an improved form of RNN. The RNN parameter update uses data from the gradients. The parameter updates lose significance as the gradient shrinks, indicating that no significant learning has taken place. Deep learning models employ LSTM, which features feedback connections.It is known for LSTM to analyze lengthy and complex data sequences. An LSTM is made up of a cell and three gates: an input gate, an output gate, and a forget gate. Given that cells retain data for a longer period of time, the gates are used to regulate data flow within the network. It decides which data should be stored, which data should be transported across the network, which data should be deleted, and which data should be forgotten. When processing, predicting, or categorization are necessary, LSTM are used. The LSTM stores data for each time step in a memory cell. Thus, the caption is produced using the LSTM.

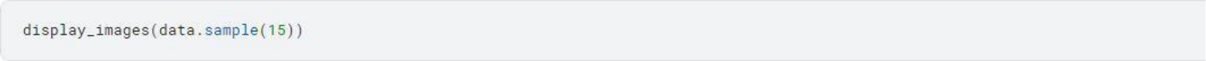
Chapter 04: RESULTS

4.1 DISCUSSION AND RESULTS

Image captioning system is a broad project with its uses in a wide variety of sectors. The main motivation for working on this project comes from the recent high development in the area especially focusing around the visually impaired or people with poor vision and the applications dedicated to them by several other big tech companies like microsoft and google. AI is currently taking over the world as even for the smallest jobs the computer can give accurate answers and results without making much of an effort. Such a convenient lifestyle should be accessible to everyone. Image captioning is a revolutionary idea in this era of social media, where people want their type of content to be readily available to them without searching for it and it coming on their feed, in such situations image captioning turns out to be the core technology. Billions of images are present on the internet, searching for a specific one out of them becomes a rigorous task. To make things easier, an image captioning system is a means to lessen the workload so the outcome of our model of image captioning generator is extremely important to show the system's working, performance and value. As discussed before in this model we use the deep learning technologies of CNN and LSTM, while there are several other approaches like RNN and CNN, human vision using hard tension and soft tension, use of YOLOv4 and many others for different modules present in the project. The reason why CNN and LSTM was chosen was because it is easy to implement and understand.The data we have chosen for are model does not have any specific people or location in it and the five captions along with it makes it easier for our model to learn how to generate captions.First the data from the dataset is read and displayed in the form of image name and caption in the output as shown in Figure 4.1, then the input images from the dataset is plotted as shown in Figure 4.2.



```
def readImage(path,img_size=224):
    img = load_img(path,color_mode='rgb',target_size=(img_size,img_size))
    img = img_to_array(img)
    img = img/255.

    return img

def display_images(temp_df):
    temp_df = temp_df.reset_index(drop=True)
    plt.figure(figsize = (20 , 20))
    n = 0
    for i in range(15):
        n+=1
        plt.subplot(5 , 5, n)
        plt.subplots_adjust(hspace = 0.7, wspace = 0.3)
        image = readImage(f"../input/flickr8k/Images/{temp_df.image[i]}")
        plt.imshow(image)
        plt.title("\n".join(wrap(temp_df.caption[i], 20)))
        plt.axis("off")
```



```
display_images(data.sample(15))
```

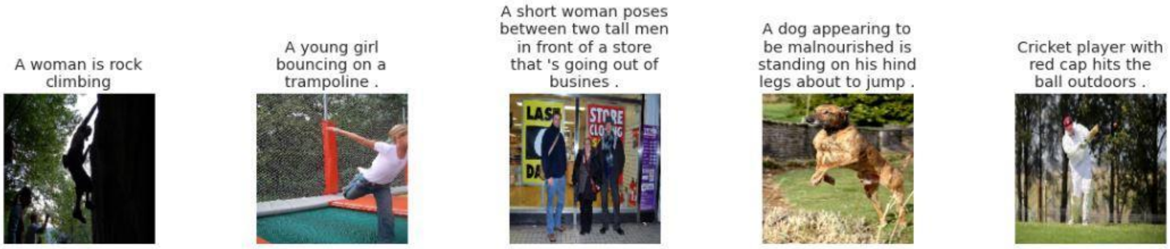

A woman is rock climbing
A young girl bouncing on a trampoline .
A short woman poses between two tall men in front of a store that 's going out of busines .
A dog appearing to be malnourished is standing on his hind legs about to jump .
Cricket player with red cap hits the ball outdoors .

Figure 4.1 The data is read

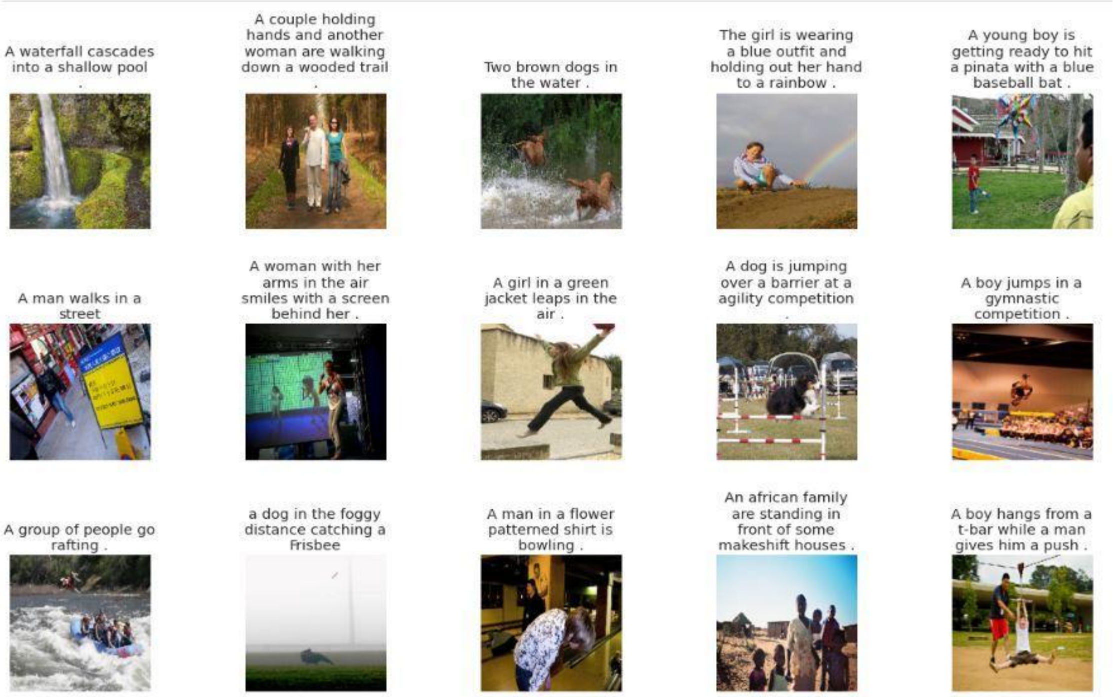
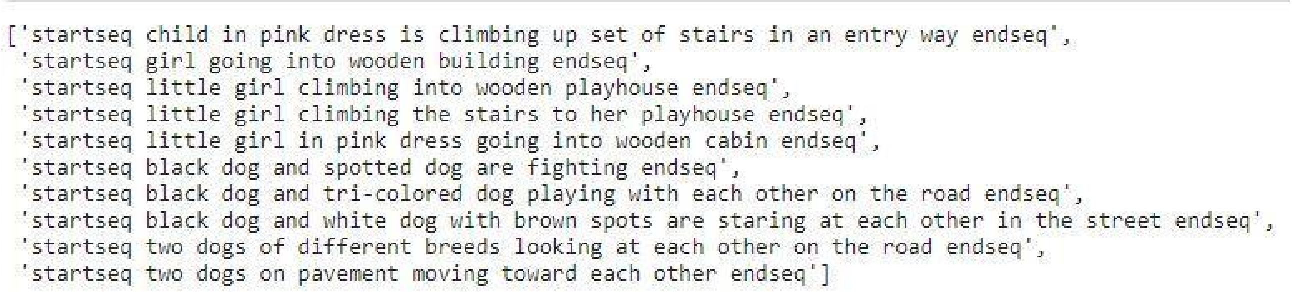

A waterfall cascades into a shallow pool .
A couple holding hands and another woman are walking down a wooded trail .
Two brown dogs in the water .
The girl is wearing a blue outfit and holding out her hand to a rainbow .
A young boy is getting ready to hit a pinata with a blue baseball bat .
A man walks in a street
A woman with her arms in the air smiles with a screen behind her .
A girl in a green jacket leaps in the air .
A dog is jumping over a barrier at a agility competition .
A boy jumps in a gymnastic competition .
A group of people go rafting .
a dog in the foggy distance catching a Frisbee
A man in a flower patterned shirt is bowling .
An african family are standing in front of some makeshift houses .
A boy hangs from a t-bar while a man gives him a push .

Figure 4.2. Sample Data, Images in the Flickr8k Dataset when model is being trained

Data text processing is done by putting the caption data in a list. Two tokens named startseq and endseq are put at the beginning and end of every caption to help in determining the start of our caption and end of it.

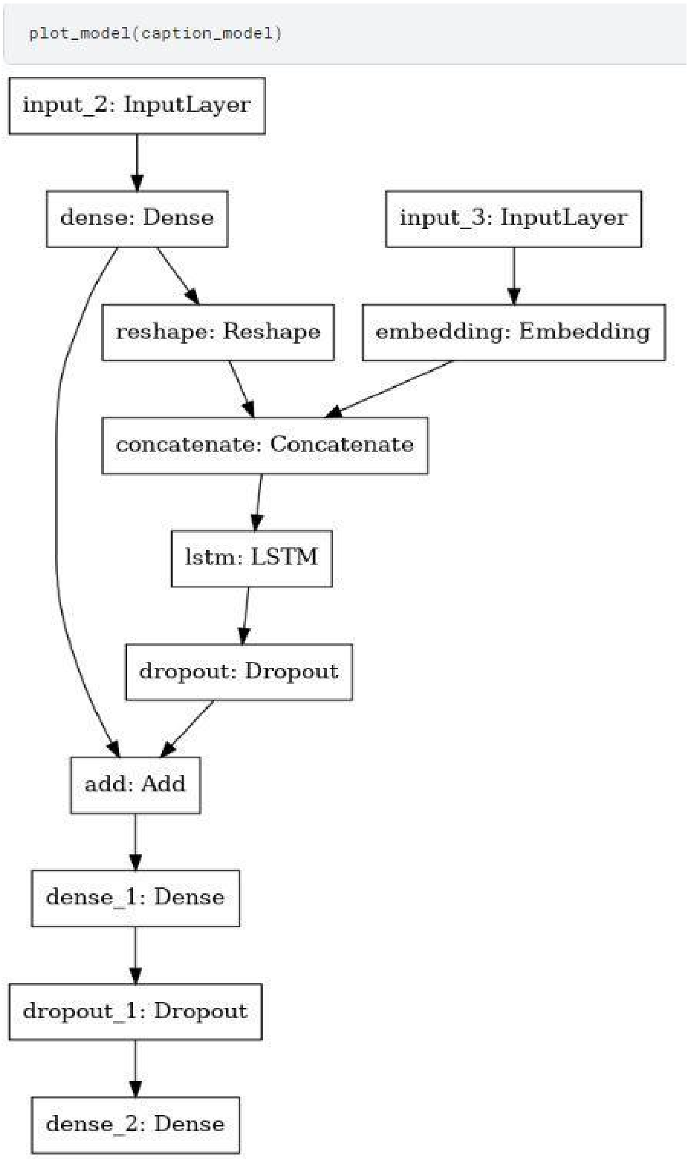


```
['startseq child in pink dress is climbing up set of stairs in an entry way endseq',
 'startseq girl going into wooden building endseq',
 'startseq little girl climbing into wooden playhouse endseq',
 'startseq little girl climbing the stairs to her playhouse endseq',
 'startseq little girl in pink dress going into wooden cabin endseq',
 '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']
```

Figure 4.3. Caption processing done with the token in the front and end

The pre-processed data is forwarded to our image based model where the input images are taken in as 2D matrix and features are extracted from the images. Then the output of this module is further fed to the LSTM network, where the image embedded representations are concatenated with the first word of the sentence that is the startseq token.

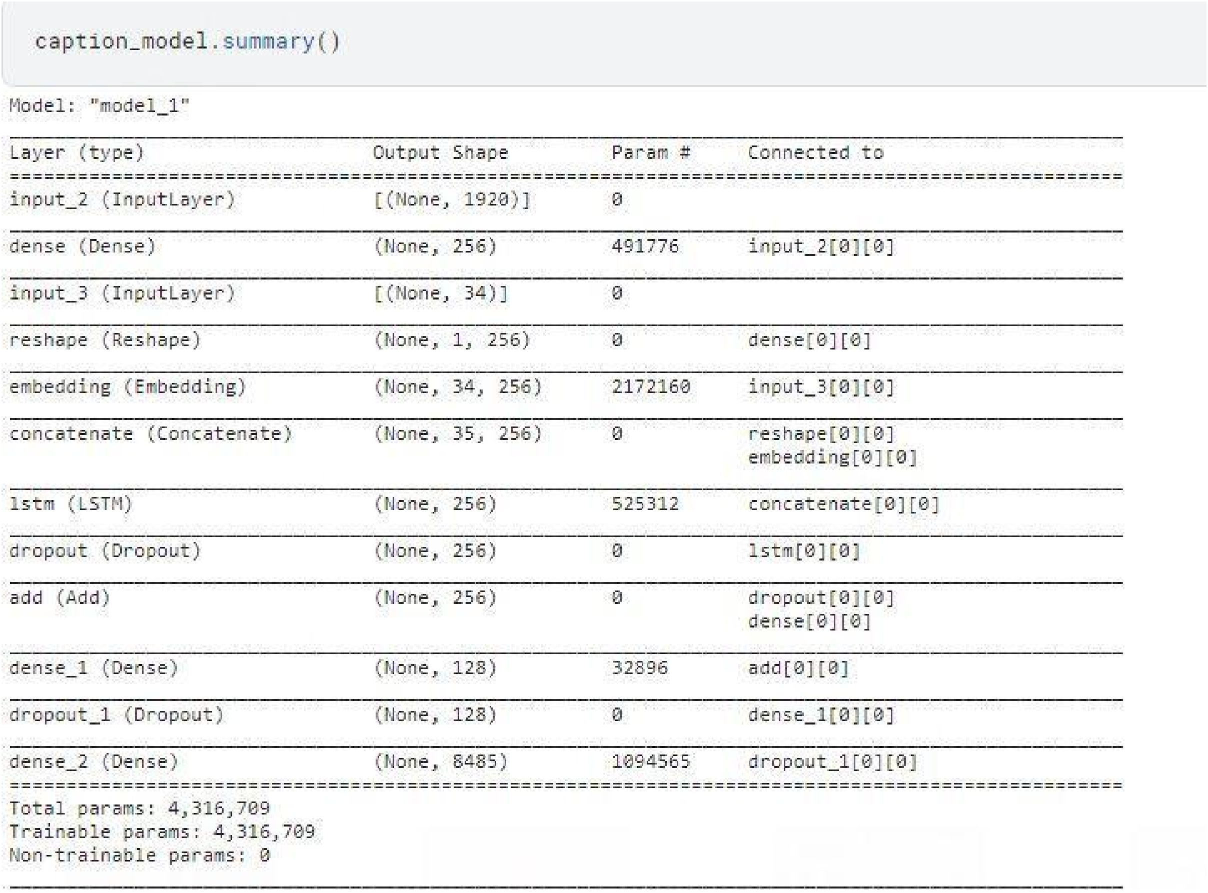
The LSTM network starts generating words after each input thus forming a sentence at the end. The caption model is displayed in Figure 4.4 after which our system is trained with the given data.Fifteen random samples are taken for caption prediction and the output is displayed.



```
plot_model(caption_model)
```

Figure 4.4. Caption Model

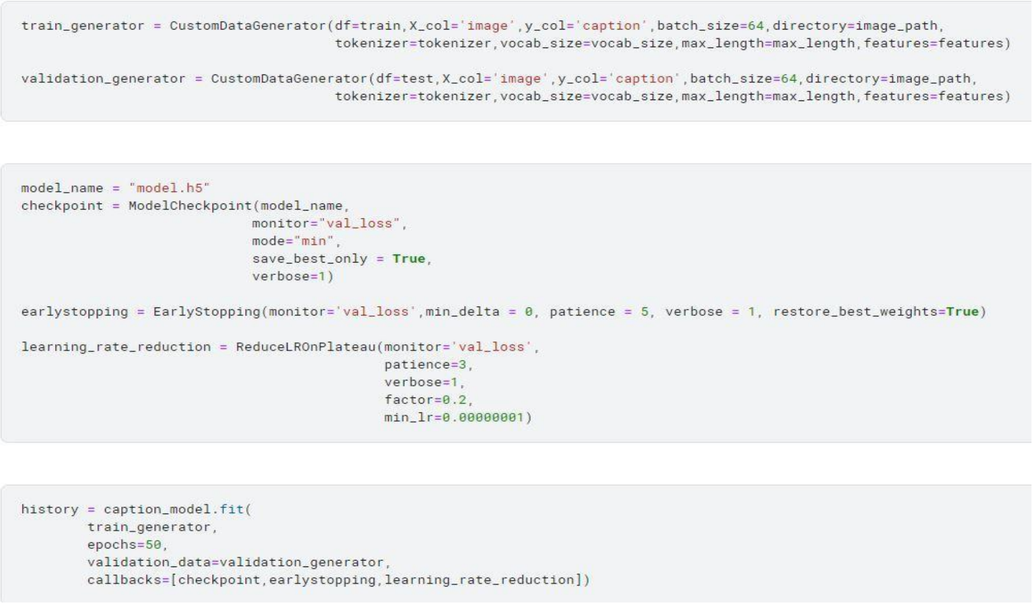
The summary of the model as well as the how the model is trained using different epochs is given in the figures below. Dense refers to the actual network layer in our model. It is used to feed all outputs from the previous layer to the neurons present in this layer, each neuron then feeds the output to the next layer. The first dense layer has 256 neurons, the second dense layer has 128 neurons. Dropout neural networks refers to the regularization technique in which we drop out some of the nodes from our network. This is done to reduce the problem of overfitting, in this layer some of the nodes are ignored or temporarily deactivated, these nodes are not fed forward and no weight updates are applied to these neurons.



```
caption_model.summary()
```

```
Model: "model_1"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_2 (InputLayer)            [(None, 1920)]       0
__________________________________________________________________________________________________
dense (Dense)                   (None, 256)          491776      input_2[0][0]
__________________________________________________________________________________________________
input_3 (InputLayer)            [(None, 34)]         0
__________________________________________________________________________________________________
reshape (Reshape)               (None, 1, 256)       0           dense[0][0]
__________________________________________________________________________________________________
embedding (Embedding)           (None, 34, 256)      2172160     input_3[0][0]
__________________________________________________________________________________________________
concatenate (Concatenate)       (None, 35, 256)      0           reshape[0][0]
                                                                 embedding[0][0]
__________________________________________________________________________________________________
lstm (LSTM)                     (None, 256)          525312      concatenate[0][0]
__________________________________________________________________________________________________
dropout (Dropout)               (None, 256)          0           lstm[0][0]
__________________________________________________________________________________________________
add (Add)                       (None, 256)          0           dropout[0][0]
                                                                 dense[0][0]
__________________________________________________________________________________________________
dense_1 (Dense)                 (None, 128)          32896       add[0][0]
__________________________________________________________________________________________________
dropout_1 (Dropout)             (None, 128)          0           dense_1[0][0]
__________________________________________________________________________________________________
dense_2 (Dense)                 (None, 8485)         1094565     dropout_1[0][0]
==================================================================================================
Total params: 4,316,709
Trainable params: 4,316,709
Non-trainable params: 0
__________________________________________________________________________________________________
```

Figure 4.5 The summary of the proposed model



```
train_generator = CustomDataGenerator(df=train,X_col='image',y_col='caption',batch_size=64,directory=image_path,
                                      tokenizer=tokenizer,vocab_size=vocab_size,max_length=max_length,features=features)

validation_generator = CustomDataGenerator(df=test,X_col='image',y_col='caption',batch_size=64,directory=image_path,
                                      tokenizer=tokenizer,vocab_size=vocab_size,max_length=max_length,features=features)
```

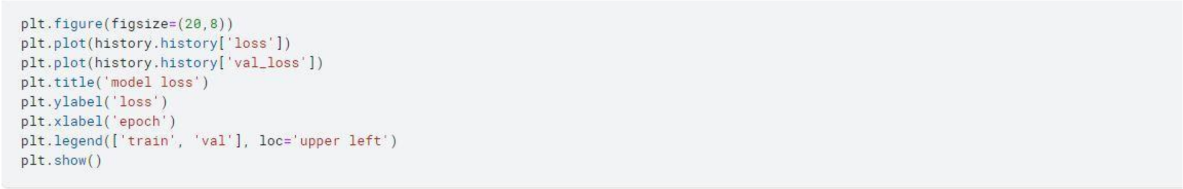
```
model_name = "model.h5"
checkpoint = ModelCheckpoint(model_name,
                            monitor="val_loss",
                            mode="min",
                            save_best_only = True,
                            verbose=1)

earlystopping = EarlyStopping(monitor='val_loss',min_delta = 0, patience = 5, verbose = 1, restore_best_weights=True)

learning_rate_reduction = ReduceLROnPlateau(monitor='val_loss',
                                            patience=3,
                                            verbose=1,
                                            factor=0.2,
                                            min_lr=0.00000001)
```

```
history = caption_model.fit(
        train_generator,
        epochs=50,
        validation_data=validation_generator,
        callbacks=[checkpoint,earlystopping,learning_rate_reduction])
```

Figure 4.6 The training of our system

A line graph can be plotted between the loss and the value loss to figure out what the model is trying to reduce.

We try to achieve the lowest loss possible in the training procedure, the loss curves run down which depicts the improvement of the model as it is training and the epochs increase. .



```
plt.figure(figsize=(20,8))
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('model loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'val'], loc='upper left')
plt.show()
```

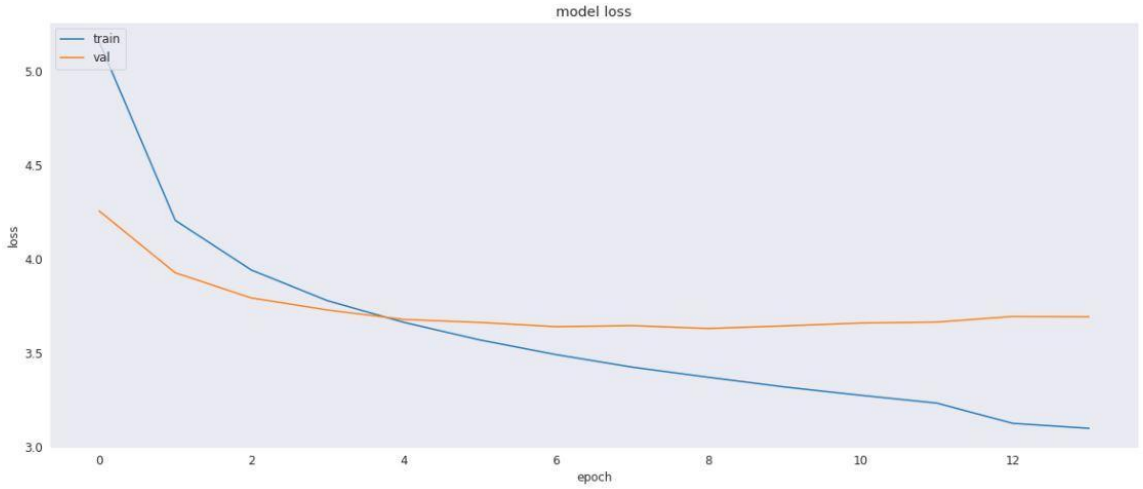


Figure 4.7 Model loss

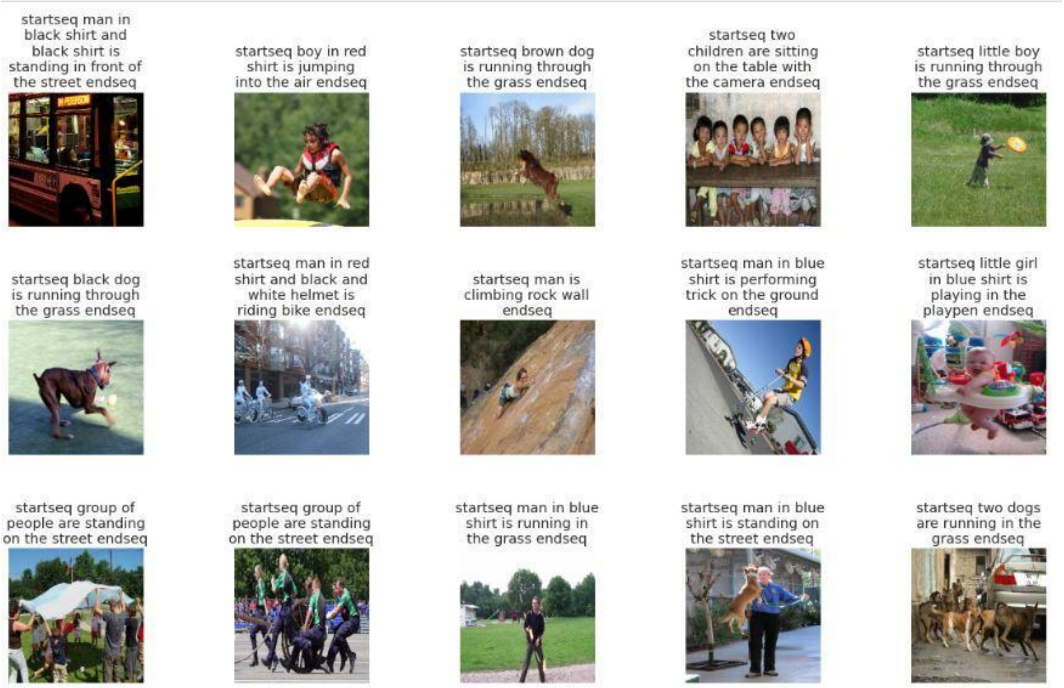

startseq man in black shirt and black shirt is standing in front of the street endseq
startseq boy in red shirt is jumping into the air endseq
startseq brown dog is running through the grass endseq
startseq two children are sitting on the table with the camera endseq
startseq little boy is running through the grass endseq
startseq black dog is running through the grass endseq
startseq man in red shirt and black and white helmet is riding bike endseq
startseq man is climbing rock wall endseq
startseq man in blue shirt is performing trick on the ground endseq
startseq little girl in blue shirt is playing in the playpen endseq
startseq group of people are standing on the street endseq
startseq group of people are standing on the street endseq
startseq man in blue shirt is running in the grass endseq
startseq man in blue shirt is standing on the street endseq
startseq two dogs are running in the grass endseq

Figure 4.8. The result of the proposed model

The end result can be seen with the two tokens at the front and end to signify the length of the caption, the model gives accurate , to the point and efficient captions of the input images that were taken randomly so we can now conclude that the model is working and can generate caption skillfully for the trained dataset.

4.2 EVALUATION

Evaluation is one of the crucial steps in projects that involves assessing the performance of a trained model. The purpose of this step is to see how well a model is making predictions on new data.

The evaluation process in our project is done by splitting the available data into training and testing data. The models are initially trained using the training data, and their performance is then assessed using the testing data. Evaluation is a key stage since it gives us important knowledge about the weaknesses, biases, and overfitting that the models may be experiencing.For the evaluation of this model we use the BLEU score.

BLEU score : It is an evaluation algorithm used for evaluating the quality of our machine generated/ translated text. BLEU score is chosen due to its various properties which include language independent, Easy to compute and understand, also since the score lies between [0,1] it is easier to compare, higher the score the more accurate is the caption.

To calculate the BLEU score first we convert the predicted caption and references to unigrams/bigrams.

N-Gram Model refers to a contiguous sequence of n items, where n can be any number like 1, 2, 3, etc., are generated from a given sample of text in natural language processing. The items can be characters or words. For the sentence “It is a sunny day today” the possible n-gram models that can be generated are:

1-grams : “It” “is” “a” “sunny” “day” “today”.

2-gram: “It is” “is a” “a sunny” “sunny day” “day today”.

3-gram: “It is a” “is a sunny” “a sunny day” “sunny day today”

4-gram: “It is a sunny” “is a sunny day” “a sunny day today”

5-gram : “It is a sunny day” “It is a sunny day today”

6-gram: “It is a sunny day today”.

For example : Predicted caption : “A dog is playing”

References : 1. A dog is chewing the ball.

2. A white dog is having fun playing

Now the bigrams for the predicted caption : (A dog) (dog is) (is playing)

Bigrams for the references

1. (A dog) (dog is) (is chewing) (chewing the) (the ball)

2. (A white) (white dog) (dog is) (is having) (having fun) (fun playing)

$$\textit{Modified n-gram precision} = \frac{maxnumber\ of\ times\ n{-}gram\ occurences\ in\ reference}{Total\ Number\ of\ n{-}grams}$$

So for the example we have:

Bleu score = $\frac{1}{3} + \frac{2}{3} + \frac{0}{3} = 1$

Chapter 05: CONCLUSION

5.1 CONCLUSION

In conclusion, this project proved to be an opportunity to understand and work on an Image captioning system. The publicly accessible datasets, information, journals and articles made it possible to work on this project efficiently and effectively in depth. Different approaches to Image captioning could be accessed and learnt about to make the decision of choosing the approach of CNN and LSTM.We have examined deep learning-based picture captioning techniques through this study. Along with broad block diagrams of the key groupings, a taxonomy of picture captioning techniques has been supplied, highlighting the advantages and disadvantages of each. We discussed several datasets and evaluation criteria, as well as the benefits and drawbacks of each. A brief summary of the experiment's findings is also provided. We went into great length on potential directions for research in this area. Despite the fact that deep learning-based picture captioning methods have advanced significantly in recent years, a dependable method that can generate high-quality captions for practically all photos has not yet been created. Automatic picture captioning will remain a prominent research area due to the creation of fresh deep learning network designs.The future of picture captioning is quite promising as more people use social media every day and the majority of them share images.The techniques used in the project, that is , CNN and LSTM are popular approaches which will be further upgraded with time and since these are easy to implement and accessible the information network will keep on growing. Since image captioning is an emerging technology we see dominating the social media along with this importance for visually impaired people to have the internet more accessible, this project serves to be a major learning achievement. Overall the Image captioning system is an important and effective advanced technology making the internet more efficient and easier to access.

5.2 APPLICATION OF MAJOR PROJECT

There are several practical applications that can be used, some of these are as mentioned below:

- Image captioning can be used in image tagging for photo sharing websites, and catalogs: When a user uploads an image to an online catalog, the image captioning system recognises the image and generates attributes like signatures, categories, or descriptions. This simplifies the user's life. It can be used to generate tags automatically. It may choose the style, fabric, color, pattern, and fit of clothing for online retailers.

- Image annotations for visually impaired people : The image can be converted to text and then to voice, both of which come under the domain of deep learning.Microsoft has previously created a smartphone software that can read text when the camera is directed at it and provides audio cues in order to assist those with vision impairments in navigating the environment. For the benefit of blind individuals, Google has also developed a comparable tool that can provide a text description for any photograph it takes.

- Image Captioning for Social Media Sites : Social media platforms like Twitter, Facebook and Instagram use this technology to help create a feed for the users according to their searches showing images with similar captions to what they searched. Five years ago, Facebook made an investment in this technology and developed a system that could generate alternative text descriptions. Over time, this system has become increasingly accurate, providing thorough descriptions of an image.

- Image captioning used for Logo Identification : Image captioning is also being used in AI app development, DeepLogo is an app used to recognize logotypes , the logos that are identified appear as the caption on the image. This makes it easier to find out a certain logo and similar logos to it in shape and category.

5.3 LIMITATIONS

There are certain limitations to Image Captioning especially errors which occur due to several reasons.

1. Common errors are due to poor image quality or incomplete dataset. The image captioning model is trained on a certain general dataset so anything out of its scope becomes tougher to identify correctly.

   a. Poor Quality or Cropped Images b. Unusual Images

2. Images and Texts are two different types of data fed into the model and they have different characteristics and representations like color, syntax , pixel values etc. Often relationships between text and images are many-to-many rather than just one-to-one so it becomes difficult for the image captioning system to effectively work properly on the types of data together as well as align them in a way that the information that is extracted is as accurate as possible.

3. Data Diversity and Quality proves to be a major factor in determining the quality of our image captioning system. Many existing datasets present are ones that are focused on a certain domain, style , are limited in size or lack complex events. The captions of images are simple, repetitive and sometimes inaccurate. As our system is trained on these specific datasets it becomes important to have large, diverse and accurate datasets to get the best possible results.

4. The evaluation metrics currently existing are BLEU, CIDEr, ROUGE and Recall@K which evaluate the dataset based on the word overlap and does not capture the depth or pragmatic aspect of the caption, which means even though the caption might be grammatically correct it can be less informative or engaging. As these metrics do not give any feedback on the generated captions or the reason for generating such captions it might affect the credibility of the system. Thus there’s a need for a more well detailed and diverse evaluation method.

5. The model does not perform well on new domains, tasks and scenarios, it does not give accurate results for unseen images or texts especially if the input data is too different to the training data. A

model trained on modern medical data related to medications and operations may not be able to guess the caption and outcome for the historical and traditional methods.

5.4 FUTURE WORK

The field of Artificial Intelligence is bound to grow more and more in the future, the field of Image captioning will remain an important technology to work towards. For further work the model can be made more accurate or faster and efficient with either using the semantic keywords framework or YOLOv4 along with Xception. Text reading models can be added to the image captioning system to help give more efficient and accurate results, which can be used in even the education sector to teach certain pronunciation along with the image. More specific dataset can be worked upon , for example for a celebrities dataset , face recognition can be used along with the image captioning system to recognize a certain celebrity from the image. The project has boundless possibilities to work with.

REFERENCES

[1] Frank Blaauw, Ando Emerencia, “Deep Learning the Beautiful Mind”, Accessed : 16 May 2023[Online]. Available:https://mindwise-groningen.nl/deep-learning-the-beautiful-mind/

[2] Abisha Anto Ignatious.L , Jeevitha.S, Madhurambigai.M, Hemalatha.M. “Semantic Driven CNN-LSTM Architecture for Personalized Image Caption Generation”, 2019 11th International Conference on Advanced Computing (ICoAC),IEEE,2020.[Online] Available :https://doi.org/10.1109/ICoAC48765.2019.246867

[3] Anish Banda, Harshavardhan Manne, Rohan Garakurthi. “Image Captioning using CNN and LSTM”, International Journal for Research in Applied Science and Engineering Technology, IJRASET, 2021.[Online] Available : http://dx.doi.org/10.22214/ijraset.2021.37846

[4] M.Pranay Kumar, V.Snigdha, R.Nandini, B.Indira Reddy. “Image Captioning Generator using CNN and LSTM”, International Journal for Research in Applied Science and Engineering Technology,IJRASET,2022.[Online] Available: http://dx.doi.org/10.22214/ijraset.2022.44502

[5] Al-Malla, Assef Jafar, Nada Ghneim. “Image Captioning Model using Attention and Object Features to Mimic Human Image Understanding”, Journal of Big Data,2022.[Online] Available : https://doi.org/10.1186/s40537-022-00571-w

[6] Priyanka Raut, Rushalo A Deshmukh. “An Advanced Image Captioning using Combination of CNN and LSTM”, Turkish Journal of Computer and Mathematics Education,2021.[Online] Available : https://doi.org/10.17762/turcomat.v12i1S.1593

[7] V.Varshith Reddy , Y.Shiva Krishna, U.Varun Kumar Reddy, Shubhangi Mahule. “Gray Scale Image Captioning using CNN and LSTM”, International Journal for Research in Applied Science and Engineering Technology,IJRASET,2022.[Online]
Available: http://dx.doi.org/10.22214/ijraset.2022.41589