BVB Campus, Vidyanagar, Hubballi – 580031, Karnataka, INDIA.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Project report on

Visual Question Answering

Submitted

in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted By

Pooja Doddannavar	01FE18BCS141
Prasann Kshirasagar	01FE18BCS149
Prasad V Patil	01FE18BCS148
Prashant Kumar	01FE18BCS150
Abhay Ambeakr	01FE18BCS006

Under the guidance of

Dr. Sujatha C

Designation

KLE Technological University
Creating Value
Leveraging Knowledge
BVB Campus, Vidyanagar, Hubballi – 580031, Karnataka, INDIA.

School of Computer Science and Engineering

KLE Technological University, Hubballi

2020-2021

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

2020-21

CERTIFICATE

This is to certify that project entitled "Visual Question Answering" is a bonafied work carried out by the student team Pooja Doddannavar 01FE18BCS141, Prasann Kshirasagar 01FE18BCS149, Prasad V Patil 01FE18BCS148, Prashant Kumar 01FE18BCS150, Abhay Ambekar 01FE18BCS006, in partial fulfillment of the completion of 7th semester B. E. course during the year 2020 – 2021. The project report has been approved as it satisfies the academic requirement with respect to the project work prescribed for the above said course.

Dr. Sujatha C
(write guide name)

SoCSE
Head
Dr. Meena S. M.

External Viva-Voce

Name of the examiners Signature with date

1 _____ _____

2 _____ _____

ABSTRACT

VQA (Visual Question Answering) is an AI-compliant job that combines computer vision (CV) and natural language processing (NLP).Problems in the junction of vision and language are important both as research questions and as a result of the wide range of applications they offer.Language and vision problems such as image captioning and visual question answering (VQA) have gained popularity in recent years as the computer vision research community is progressing beyond recognition and towards solving multi-modal problems.However, recent research has shown that language can give a strong prior that can result in good surface performance even when the underlying models do not fully comprehend the visual content.

In this paper, we present a VQA model that combines two modules: one that uses InceptionV3 to encode both picture and text simultaneously for caption generation, and the other that uses the Multitask QA QG module for answer extraction, question generating, and question answering. We aim to create a caption based on the visual features, then create a question based on the caption, and finally answer it.

Keywords : *Visual Question Answer, Caption.*

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of a number of individuals whose professional guidance and encouragement helped me in the successful completion of this report work.

We take this opportunity to thank Dr. Ashok Shettar, Vice-chancellor and to Dr. N H Ayachit, Registrar, KLE Technological University, Hubballi.

We also take this opportunity to thank Dr. Meena S M, Professor and Head of Department, Department of Computer Science and Engineering for having provided us academic environment which nurtured our practical skills contributing to the success of our project.

We sincerely thank our guide Dr. Sujatha C, Professor, Department of Computer Science and Engineering for his guidance, inspiration and wholehearted co-operation during the course of completion.

We sincerely thank our project co-ordinators Dr. S. G. Totad for his support, inspiration and wholehearted co-operation during the course of completion.

Our gratitude will not be complete without thanking the Almighty God, our beloved parents, our seniors and our friends who have been a constant source of blessings and aspirations.

Pooja Doddannavar - 01FE18BCS141
Prasann Kshirasagar - 01FE18BCS149
Prasad V Patil - 01FE18BCS148
Prashant Kumar - 01FE18BCS150
Abhay Ambekar - 01FE18BCS006

CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

The task of answering questions concerning a specific piece of visual content, such as an image, video, or infographic, is known as visual question answering (VQA). Recognizing entities and objects, reasoning about their spatial and temporal interactions, reading text, parsing audio, interpreting abstract and graphical illustrations, and using external knowledge not directly present in the given content are just some of the skills required to answer questions about visual content. VQA has recently been a major topic in the fields of computer vision, natural language understanding, and artificial intelligence.

Answering visual questions necessitates the acquisition of daily common knowledge and the modelling of the semantic relationship between different components of images, which is too complex for VQA systems to learn from images with only responses as guidance.Meanwhile, to address this problem, we present a system that can produce image captions and answer visual questions by combining two tasks that compensate for each other. In particular, we leverage image features to generate question-related captions and use the generated captions as additional features to provide new knowledge to the VQA system when it generates questions and then answers them.

1.1 Motivation

Visual Question Answering (VQA) is a difficult method to implement, although it is simple for humans. To answer a question regarding a given image, humans blend visual data with general and commonsense knowledge.The difficulty of adding general knowledge into VQA models while using visual input is addressed in this study. We provide a model that captures the visual scene aspects in the system's text-based input and output.

The task of open-ended Visual Question Answering is introduced in this work (VQA). A VQA system takes an image and a free-form, open-ended, natural-language query about it as inputs and outputs a response. This goal-oriented activity can be used in situations where visually impaired people or intelligence analysts are actively eliciting visual data.

1.2 Literature Survey

Zhang, Shijie[1] proposed the model which is able to generate visually grounded questions with diverse types for a single input image. Their model takes images as an input and samples the most probable question types, and generates the questions in sequel. They start with randomly picking a caption from a set of automatically generated captions, which describes a certain region of image with natural language. Then sampling a reasonable question type with varying caption. In the last step, the question generator learns the correlation between the caption and the image, generating a question of the chosen type. Formally, for each raw image x, their model generates a set of captions c1, c2, ..., cm, samples a set of question types t1, t2, ..., tm , followed by yielding a set of grounded questions q1, q2, ..., qm . Herein, a caption or a question is a sequence of words, w = w1, ..., wm are generated.

Mora, Issey Masuda[2] have proposed a model that can generate Question and answer pairs given an image. The visual features of the image are extracted using the VGG-16 Net. Later, these features are injected to Long Short-Term Memory RNN, which will learn how to generate an embedding of the question. Then these embeddings are given to another LSTM which will produce an answer for the question.

The authors Jialin, Zeyuan Hu, and Raymond J. Mooney[3] have generated VQA by training an existing caption dataset, which automatically determines question relevant captions using online gradient based method. Here human annotated captions are being used and for that they have achieved a score of 59.6For image captioning a CNN module is used which takes input features and learns attention weights to predict words at each step. For generation of VQA the top- down attention features are used to get the relative answer for the question. For image and question embedding GRU(Gated Recurrent Unit) Networks are being used.

In this paper[4] the authors have discussed about Visual Question Answering (VQA) model in the medical domain exploring the approaches of clinical decision support. They introduced a VQA-RAD dataset, the first manually constructed dataset where clinicians asked naturally occurring questions about radiology images and provided reference answers. Their proposed model includes two well-known VQA methods: Multimodal Compact Bilinear pooling (MCB) and Stacked Attention Network (SAN) to generate question and answers for radiology images.

Authors in [5] this paper have proposed a model for Visual Question Generation(VQG). The work in this paper focus on developing the capability to ask relevant and to-the-point questions. Where given an image, the system is tasked with asking a question. Three distinct datasets, namely VQG COCO, VQG Bing and VQG Flicker, each covering variety

of images was used. Also three generative models, Maximum Entropy Language Model, Sequence2Sequence model and Gated Recurrent Neural Network are used to support the process of VQG.

1.3 Problem Statement

We propose to develop a Document Visual Question Answering model which generates question and answer based on the input document image.We generate the Captions initially and then generate the Questions and Answers based on the Captions.

1.4 Objectives

- To Extract the visual features from the images and generate the captions.
- To generate the questions from the captions generated.
- To predict the answers of the questions based caption and input question.

Chapter 2

REQUIREMENT ANALYSIS

Requirement's analysis focuses on the tasks that determine the requirements or conditions to satisfy the new or altered product or project, taking account of the possibly conflicting requirements of the various participants, analyzing, documenting, validating, and managing software or system requirements.

2.1 Functional Requirements

Functional requirements define the fundamental system behavior for the Masked Face Recognition System. It defines a function of the system or its component, where a function is described as a specification of behavior between inputs and outputs. The plan for implementing functional requirements is detailed within the system design.

- The system shall be able to have an unbiased, over 8000+ images in the training and testing phase.
- The system shall be able to captioned the non-captioned images.
- The system shall be able to generate the caption from the extracted feature of the image.
- The system shall be able to generate the question and answer from the generated caption.
- The system shall be able to generate the valid question and answer.

2.2 Non Functional Requirements

A non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. The plan for implementing non-functional requirements is detailed in the system architecture.

- The system should be portable and can be applied to embedded devices with limited computational capacity.
- The system should be user friendly and self-descriptive for maintenance purposes.

- The system response time should be less than 5 seconds per input image for the generation of caption.
- The system should generate the question within 3 seconds after the given input answer.

2.3 Hardware Requirements

- Processor: 1.6 GHz Intel Core i5/Pentium IV 2.4 GHz.
- RAM: at least 2 GB RAM..
- Speed: 500 MHz.
- Hard Disk: 80 GB minimum.
- Accessories: A high quality wireless/webcam camera, LCD/LED .
- Monitor, Keyboard, Mouse.

2.4 Software Requirements

Operating System:

- Windows 8 or later
- Mac OS 10.13.6 or later (preferable)
- Ubuntu 16.04 or later (64-bit) (preferable)
- PyCharm/ VSCode editor/ TensorFlow GPU (optional)

Programming Language:

- Python (3.7.6)

Open Libraries:

- openCV (Intel's Computer Vision Open-Source Library) (4.2.0)
- TensorFlow (1.14.0)
- keras (2.3.1) (TensorFlow backend)
- sklearn (0.22.1)
- imutils (0.5.3)

- numpy (1.18.2)
- matplotlib (3.1.3)
- argparse (1.1)

Chapter 3

SYSTEM DESIGN

In this chapter we discuss the framework for Caption generation using image encoding and text encoding and masked Question answer generation using T5 model. The proposed high-level design and its workflow is explained along with dataset generation and dataset pre-processing.

3.1 Workflow of the proposed system

In this system first, we will take an input image,then the image is passed through caption generator in which image encoding and text encoding takes place and captions will be generated from the caption generator, now question and answer will be generated from the question-and-answer generator model which is trained with finetuned t5 encoder and decoder.

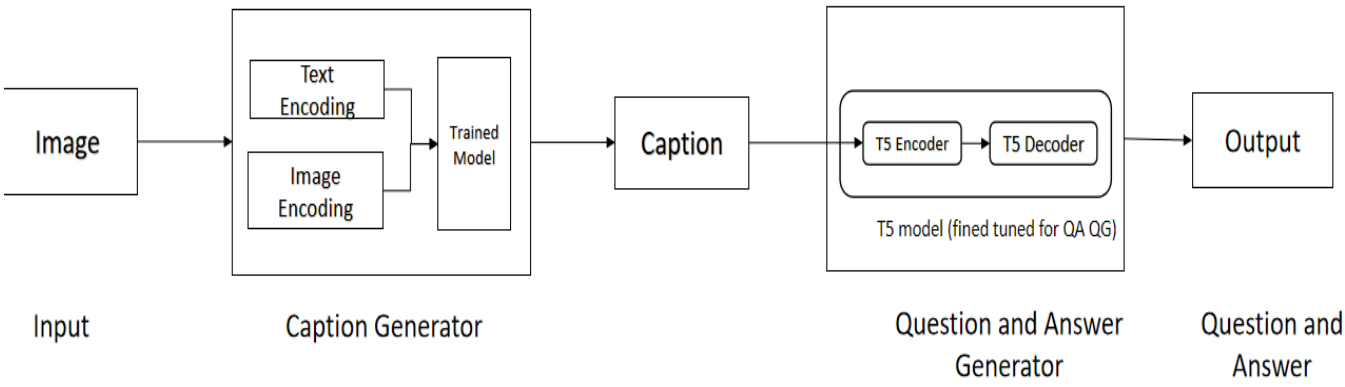


Figure 3.1: Framework for visual question answering

3.1.1 Framework for image captioning

In this module encoding of the image from CNN is done to get its feature vector.During encoding, the image is passed through various convolutional kernels which extracts the useful information from the images.For text encoding,first the text data was cleaned using NLP techniques and captions are tokenized with respect to images,then the word embedding is carried out using glove embedding model weights for word2vec representation of words to achieve the relation between the words.Then image vectors and embedded words were passed to LSTM for image captioning.The encoded images features and embedded words from LSTM are mapped with fully connected layer to generate the captions for the images. Figure 3.2 gives us the brief about the Framework of the caption generation

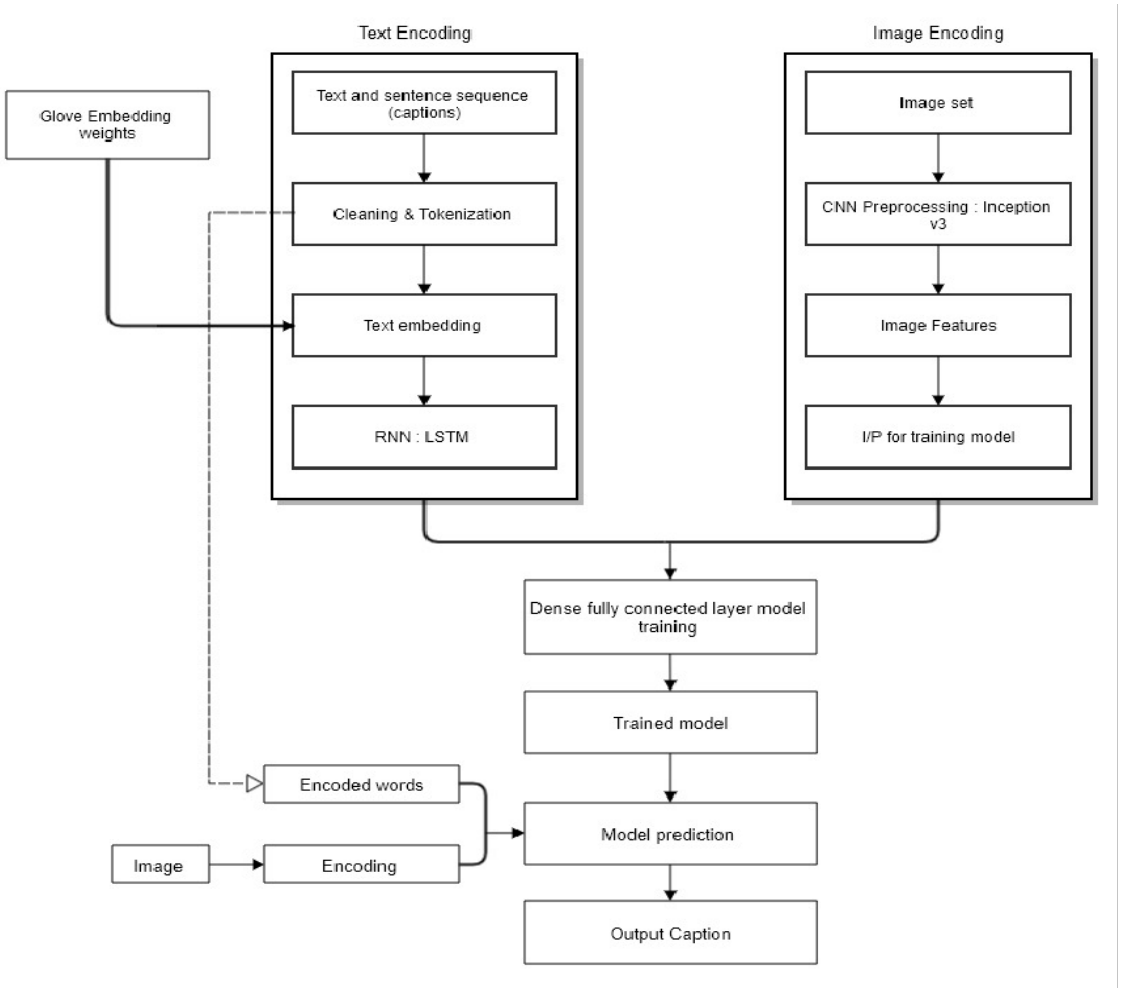


Figure 3.2: Framework for caption generation

3.1.2 Framework for Question and Answer Generation

In this module, firstly the context is given along with the answer. In the next step context+answer is feeded to the fine-tuned mt5 model which generates the question.Further to cross-check whether the generated question is accurate the model is feeded with the context+question which in turn produces the answer. If the answer generated is similar to the answer in the previous step then the generated question and answer is considered to be valid.

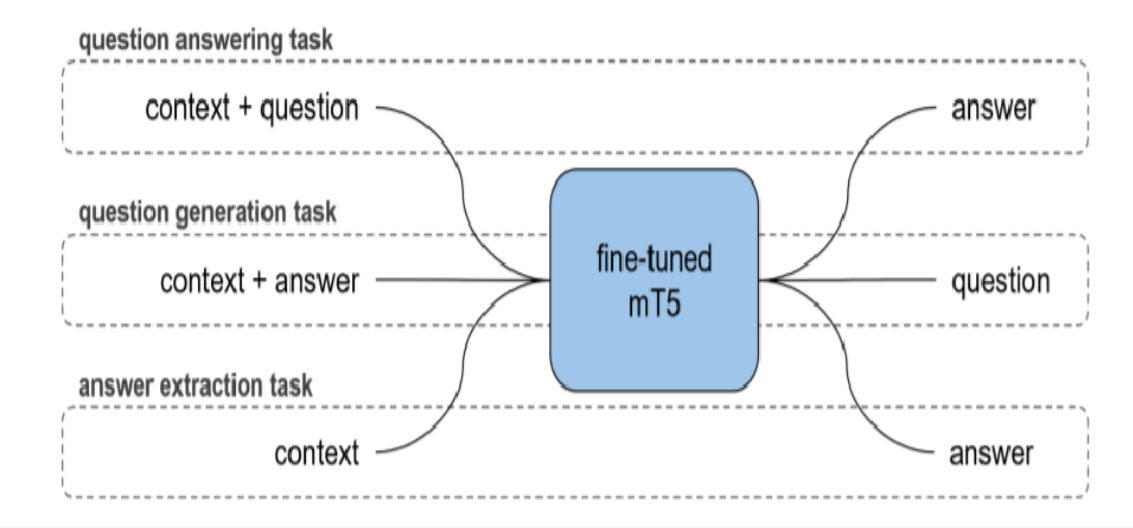


Figure 3.3: Framework for question and answer generation

Chapter 4

IMPLEMENTATION

This chapter gives a brief description about the implementation details of the system like how image captioning is performed (module 1) and the way in which these generated captions are used for question and answer generation (module 2) and which CNN architecture and transformer model is applied for building module 1 and module 2 respectively . The workflow of our proposed methodology can be seen in Figure 3.1. According to which this chapter is divided into two modules, image captioning and question and answer generation.

4.1 Image captioning

In this module, we generate relevant captions from the given input images.Here text and image encoding occur simultaneously which is used for caption generation.

The process of creating a textual description for a given image is known as image captioning. Image captioning marks images with appropriate captions ,that is it converts sequence of pixels to a sequence of words. For this purpose, processing of both the statements and images is required. For processing the language part, we use Long Short-Term Memory network (a type of recurrent neural network) and for processing image part, we use InceptionV3 (a type of CNN network) model.

The below figure 4.1 depicts the framework for image caption generation. Here we use Show and Tell model which intakes image as an input and converts it to word vector which in turn is translated to caption using LSTM network(acts as decoder).This Show and Tell model is further split up into 3 sub-sections they are, image encoder (feature extractor),text encoder(sequence processor) and decoder.

Image encoding using InceptionV3

To encode the image attributes, a better transfer learning model was needed as there are a lot of models that we can be utilized like VGG-16, InceptionV3,ResNet, etc. Among them InceptionV3 was used for image encoding as it had least number of training parameters in contrast to the other models and was robust to encode the images.

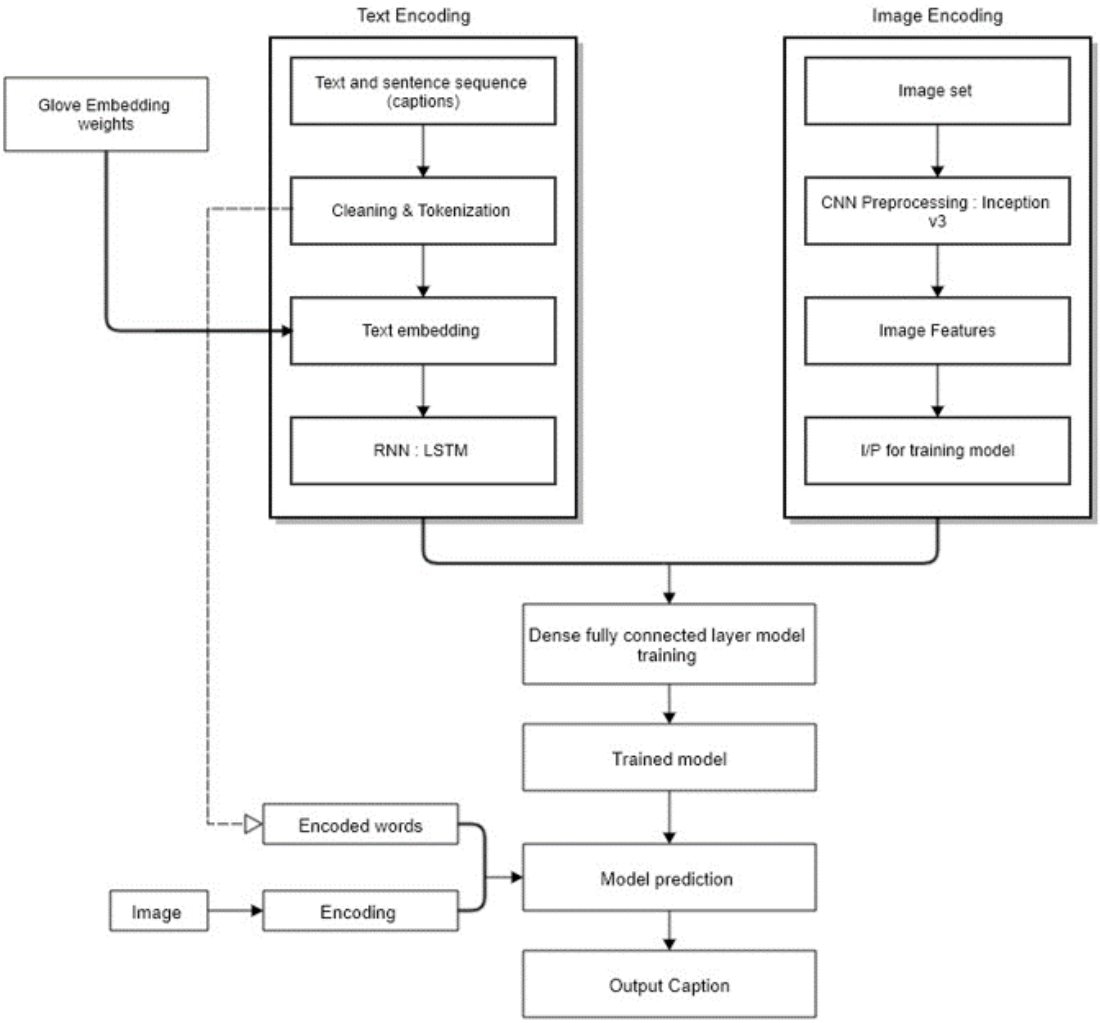


Figure 4.1: Proposed framework for image caption generation using InceptionV3 and LSTM

The proposed InceptionV3 architecture consists of repeated Inception modules, they are a combination of convolutional layers, that analyse adjacent groups of features coming as output from the previous Inception modules in the network, and pooling layers, which output a function f (in this case, max as we have applied maxpooling) of incoming inputs from the convolutional phase.Their structure is shown in Figure 4.2.

As the LSTM model requires a fixed sized vector so we had to convert every image into a fixed sized vector which can then be fed as input to the neural network. For this purpose, we opt for transfer learning by using the InceptionV3 model created by Google Research. This model was trained on Imagenet dataset to perform image classification on 1000 different classes of images. However, our purpose here is not to classify the image but just get fixed-length informative vector for each image. Hence, we just remove the last softmax layer from the model and extract a 2048 length vector (bottleneck features) for every image. we just remove the last softmax layer from the model and extract a 2048 length vector (bottleneck features)

for every image.

We'll mapeach word to a 200-dimensional vector to encode our text sequence. A glove model that has been pre-trained will be used. After the input layer, a separate layer called theembedding layer will be used to map the data. We'll use two popular ways to produce the caption: greedy search and beam search. These techniques will assist us in selecting the most appropriate words to adequately describe the image. This model was done using ImageNet dataset. The model's final layer, which is used forclassification, has been eliminated. The main goal is to generate real-time captions for each inputimage in a single pass while assuring that the result is accurate by training themachine using a good dataset.This model has been given a variety of images to test whether the model isproducing the correct output/caption for images or not.

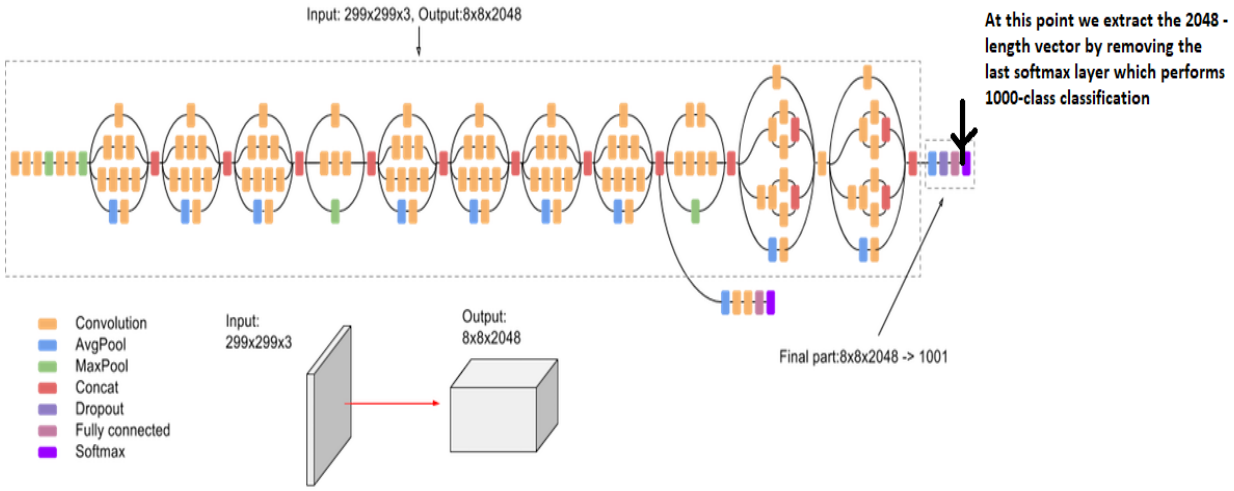


Figure 4.2: Feature Vector Extraction from InceptionV3

Text encoding using sequence processor

The prediction of the entire caption, given the image does not happen at once. We predict the caption word by word. Thus, we need to encode each word into a fixed sized vector. The input sequences are of 34 words (captions having less than 34 words are appended with 0 index),each word of the sequence is mapped to a 200-dimensional vector with the help of pre-trained glove model in order encode our text sequence. After this, a separate layer called the embedding layer is used to map the data to higher dimensional space hence producing a output of 256 element vector.

Decoder

The output of both LSTM (in case of caption model) and dense layer (in case of image model) now have the same shape of 256 element vectors. So both the input tensors are added(merged) into a single tensor using tensor addition. The result vector consisting of encoded images features and embedded words from LSTM are mapped with fully connected layer to generate the captions for the images.The below fig 4.3 depicts the workflow carried out between InceptionV3 model, LSTM model and fully connected layer. The output layer (Softmax) generates the probability distribution of all the words present in the vocabulary.

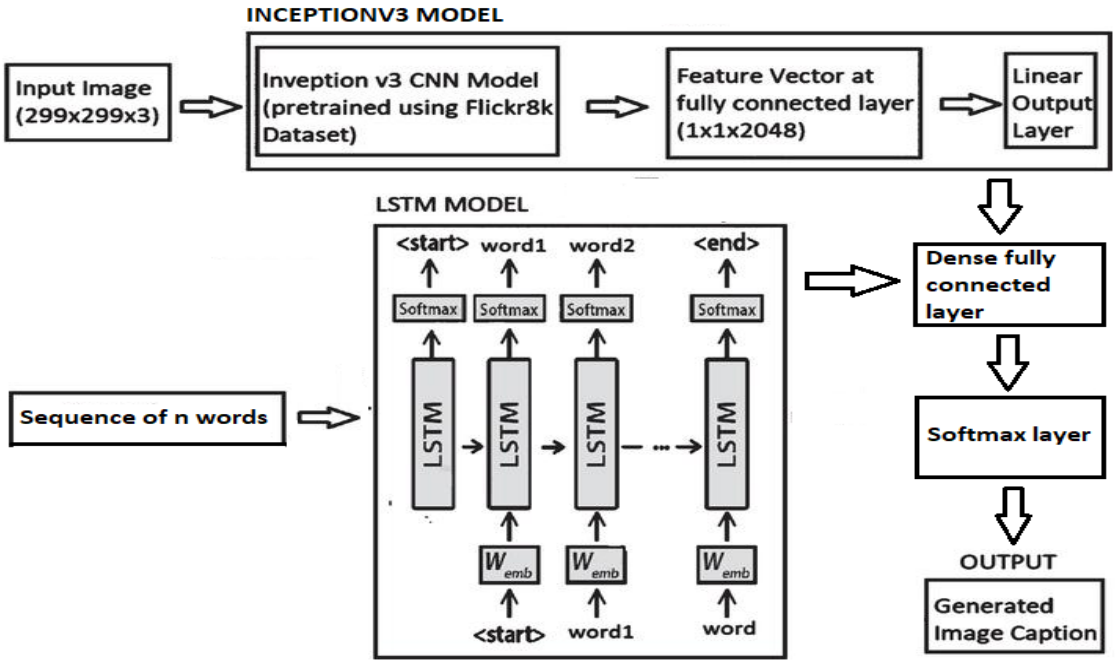


Figure 4.3: Flowchart of the proposed architecture

For defining the structure of our model, we built the image captioning module having three major steps:

- Processing the sequence from the text.
- Extracting the feature vector from the image.
- Decoding the output by concatenating the above two layers.

In the next module, these resulting captions are fed to visual question and answering model to generate relevant questions and answers.

4.2 Question and answer generation

In this module we generate the question and answer from the given input caption. This is further divided into question generation and the answer generation.

Question generation is the task of automatically generating questions from the generated captions.In the process of question generation, the model is presented with the answer and the caption and asked to generate a question for that answer by considering the caption as the context. It is transformer(t5) based model which is used for generating answer based upon given input of the context and question and it is already pretrained with squad Dataset.

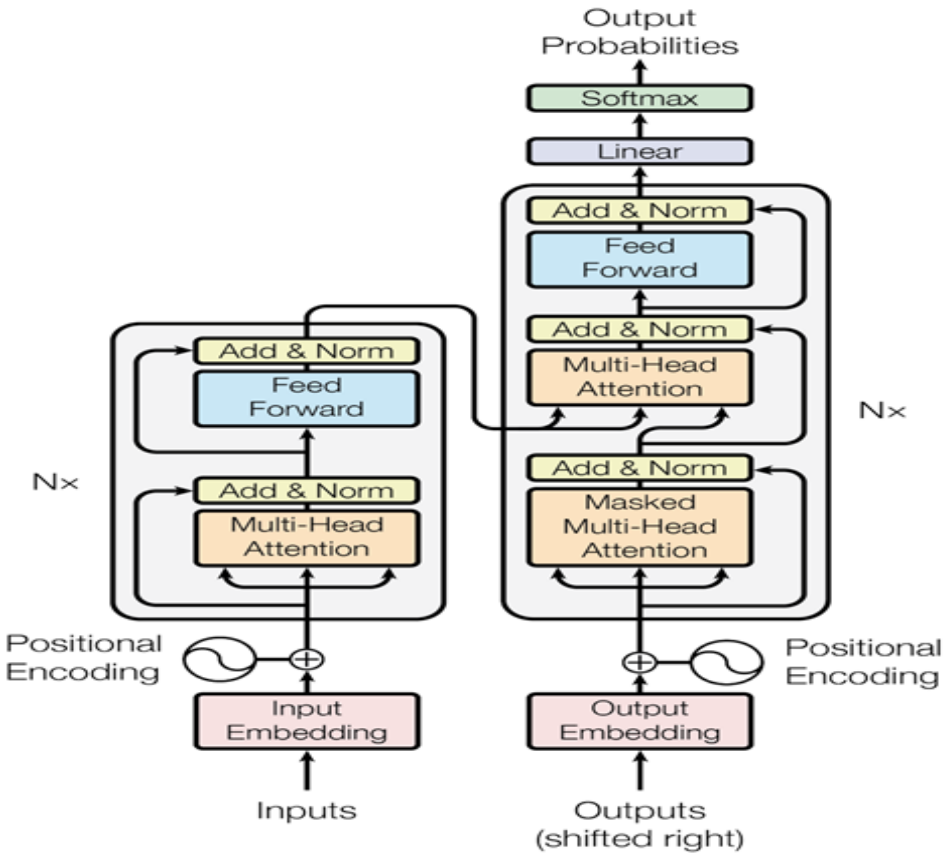


Figure 4.4: Framework for Multitask qa-qg

In figure 4.4 the caption acts as the input embedding, Once the caption is inserted, positional encoding is carried out where each word is encoded on the basis of the position according to the sentence further the caption is passed through the feed forward layer where the encoding takes place of the input caption. Once the encoding is done now the decoding process takes place.

The decoder is similar in structure to the encoder except that it includes a standard attention mechanism after each self-attention layer that attends to the output of the encoder. The below figure 4.5 depicts the workflow of the multitasak qa-qg model where firstly the context is given along with the answer. In the next step context+answer is feeded to the fine-tuned mt5 model which generates the question.

Further to cross-check whether the generated question is accurate the model is feeded with the context+question which in turn produces the answer. If the answer generated is similar to the answer in the previous step then the generated question and answer is considered to be valid.

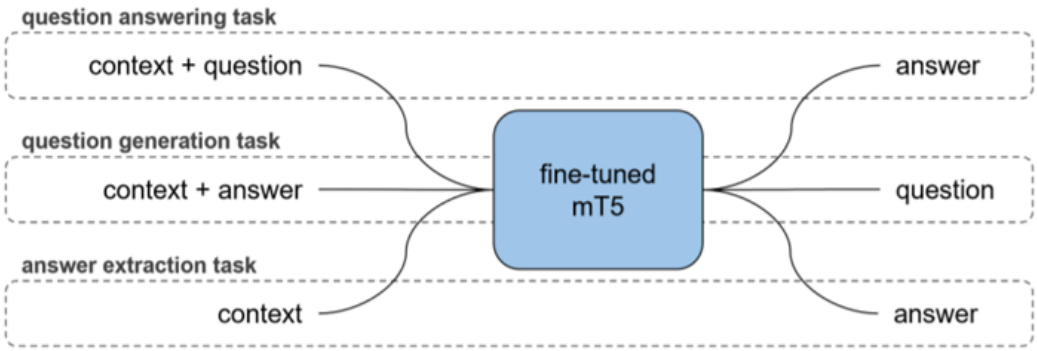


Figure 4.5: Workflow of Multitask qa-qg

This is how we have achieved the visual question answering where when to the given input is the image for which the caption is generated. Further the caption is feeded to the fine-tuned Transformer model which generates the question and the answer for the input image.

Chapter 5

RESULTS AND DISCUSSIONS

In this chapter, we will be briefing about the results of i) image caption generation using InceptionV3(image feature extraction),LSTM(for processing sequence from text) and fully connected network(for decoding output by concatenating the results of above two models) and ii) question and answer generation using mt5 transformer model finetuned for multitask qa-qg task which intakes generated captions and produces questions and answers for the same.

5.1 Dataset details

For image captioning task Flickr 8k (containing8k images) dataset provided by the University of Illinois at Urbana-Champaign was used which comprises of more than 8,092 photos and up to 5 captions for each photo.

Why Flickr8k dataset? The dataset is small in size. As a result, the model may be easily trained on low-end laptops/desktops. Data is properly labelled and for each image 5 captions are provided. The dataset is available for free. Flickr8k Dataset consists of two folders i) Flickr8k image dataset and ii) Flickr8k text dataset.

5.1.1 Flicker8k image dataset

The image dataset contains a total of 8092 JPEG images of various shapes and sizes. Out of which 6000 are for training, 1000 for testing, and 1000 for development.

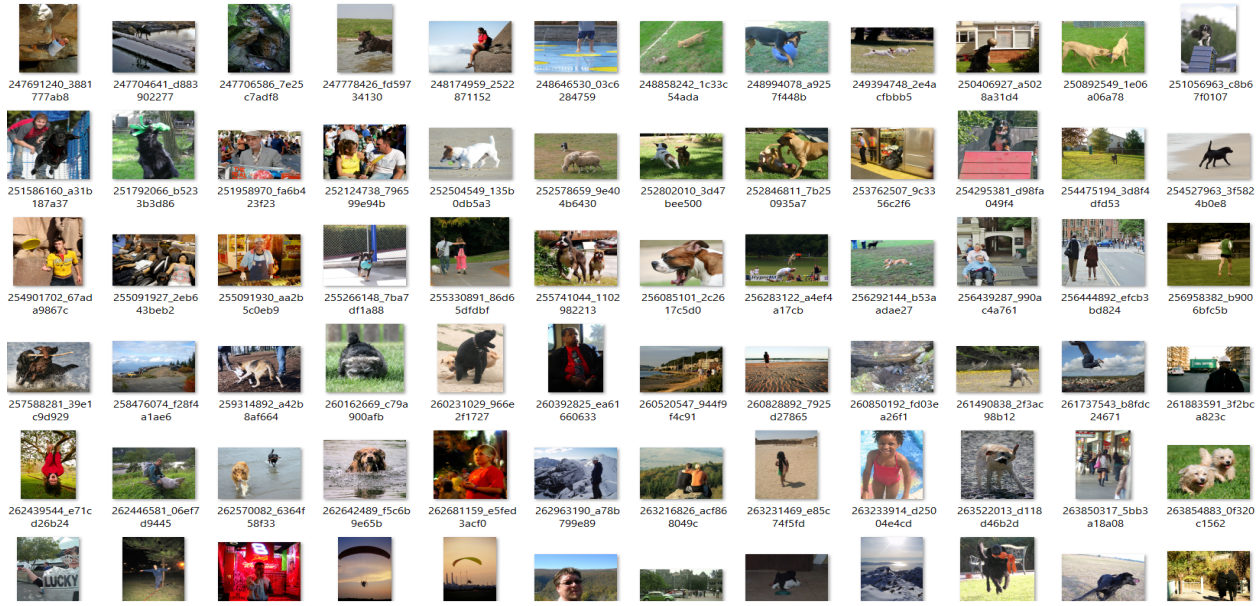


Figure 5.1: Sample image from the image dataset

5.1.2 Flicker8k text dataset

Contains text files describing train set and test set. Flickr8k.token.txt contains 5 captions for each image making a total of 40460 captions.

```
1   101654506_8eb26cfb60.jpg#0      A brown and white dog is running through the snow .
2   101654506_8eb26cfb60.jpg#1      A dog is running in the snow
3   101654506_8eb26cfb60.jpg#2      A dog running through snow .
4   101654506_8eb26cfb60.jpg#3      a white and brown dog is running through a snow covered field .
5   101654506_8eb26cfb60.jpg#4      The white and brown dog is running over the surface of the snow .
6
7   1000268201_693b08cb0e.jpg#0     A child in a pink dress is climbing up a set of stairs in an entry
8   1000268201_693b08cb0e.jpg#1     A girl going into a wooden building .
9   1000268201_693b08cb0e.jpg#2     A little girl climbing into a wooden playhouse .
10  1000268201_693b08cb0e.jpg#3     A little girl climbing the stairs to her playhouse .
11  1000268201_693b08cb0e.jpg#4     A little girl in a pink dress going into a wooden cabin .
```

Figure 5.2: Sample image from the text dataset

5.1.3 Exploratory Data Analysis Results

In order to clean the captions, removing of common words such as "a", or "the", or punctuations was needed so the three functions remove punctuation, remove single character and numeric characters were included.

```
I ate 1000 apples and a banana. I have python v2.7. It's 2:30 pm. Could you buy me

Remove punctuations..
I ate 1000 apples and a banana I have python v27 Its 230 pm Could you buy me iphone

Remove a single character word..
 ate 1000 apples and banana have python v27 Its 230 pm Could you buy me iphone7

Remove words with numeric values..
    ate         : True
    1000        : False
    apples      : True
    and         : True
    banana      : True
    have        : True
    python      : True
    v27         : False
    Its         : True
    230         : False
    pm          : True
    Could       : True
    you         : True
    buy         : True
    me          : True
    iphone7     : False
 ate apples and banana have python Its pm Could you buy me
```

Figure 5.3: Result obtained after cleaning captions

By this the vocabulary size(consisting of unique words) was reduced by 10212 words from 18975 words to 8763 words.

5.2 Evaluation Metrics

Since image captioning and question and answer generation task involves sentence generation, it needs a special kind of evaluation criteria to be followed in order to measure performance of the model. To evaluate the prediction power of our model we used BLEU (Bilingual Evaluation Understudy Score) as an evaluation parameter.

BLEU score helps to measure the accuracy of the sentence generated by the visual question and answer generation model. The score takes the range of value between 0 and 1 where the highest value of '1' indicates a perfect match between the generated caption and the actual/reference caption. Whereas '0' denotes that there is no relevancy in the generated caption with the expected caption.

A couple of ngram modified precisions are used to compute BLEU. Specifically,

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

where p_n is the modified precision for nngram, the base of log is the natural base e, w_n is weight between 0 and 1 for logp_n and $\sum_{n=1}^{N} w_n = 1$, and BP is the brevity penalty to penalize short machine translations.

$$\text{BP} = \begin{cases} 1 & \text{if } c > r \\ \exp\left(1 - \frac{r}{c}\right) & \text{if } c \leq r \end{cases}$$

where c is the number of unigrams (length) in all the generated captions, and rr is the best match lengths for each generated captions in the corpus.Here the best match length is the closest reference sentence length to the generated captions.

Usually, the BLEU is evaluated on corpus where there are many generated captions translated from different source texts and each of them has several reference captions. Then cc is the total number of unigrams (length) in all the generated captions,and rr is the sum of the best match lengths for each generated captions in the corpus.

It is not difficult to determine that BLEU is always a value between 0 and 1. It is because BP, w_n, and p_n are always between 0 and 1, and usually, BLEU uses N=4 and w_n= 1/N.

$$\begin{aligned} \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) &= \prod_{n=1}^{N} \exp\left(w_n \log p_n\right) \\ &= \prod_{n=1}^{N} \left[\exp\left(\log p_n\right)\right]^{w_n} \\ &= \prod_{n=1}^{N} {p_n}^{w_n} \\ &\in [0, 1] \end{aligned}$$

This is the combined model for the image captioning and question and answer generation so we have calculated the BLEU score of image captioning module and for question and answer generation module ,there is no ground truth given, hence we can't automatically test our model. So, we went with manual performance evaluation, using Google search and Precision@K method.

For image captioning module, the mean BLEU score comes out to be 0.374 for entire training dataset consisting of 6000 images and mean BLEU score comes out to be 0.291 for testing dataset consisting of 1000 images.

```
[69] for jpgfnm, image_feature, tokenized_text in zip(fnm_train,di_train,dt_train):
       caption_train = predict_caption(image_feature.reshape(1,len(image_feature)))
       bleu_train = sentence_bleu([caption_true],caption_train)
       bleus.append(bleu_train)
       print("Mean BLEU score for entire training set {:4.3f}".format(np.mean(bleus)))

     Mean BLEU score for entire training set 0.374
```

Figure 5.4: Mean BLEU score for entire training set.

```
[68] for jpgfnm, image_feature, tokenized_text in zip(fnm_test,di_test,dt_test):
       caption_test = predict_caption(image_feature.reshape(1,len(image_feature)))
       bleu_test = sentence_bleu([caption_true],caption_test)
       bleus.append(bleu_test)
       print("Mean BLEU score for entire testing set {:4.3f}".format(np.mean(bleus)))

     Mean BLEU score for entire testing set 0.291
```

Figure 5.5: Mean BLEU score for entire testing set.

For evaluating question and answer generation module, manual performance evaluation using Google search and Precision @ K method is carried out.

Precision: It measures the amount of instances that were identified correctly as positive and can be measured as follows:

Precision=TP/(TP+FP)

Where: TP stands for True Positives: The number of questions and answers that belongs to relevant class.

FP stands for False Positives: The number of questions and answers that belongs to irrelevant class.

Precision@K:

It measures the precision at each k level starting from k = 1 till k = n, where n is the total number of images that are used to measure the question and answer precision. We will use the Mean Average Precision (MAP) to get an estimate of the precision of our system on multiple images:

Mean Average Precision (MAP):

Where: Q: The number of images.

m_j: the total number of true positives for captions j.

$R_(j,k)$: precision of the relevant result k in captions j.

$$MAP(Q) = \frac{\sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} precision(R_{j,k})}{|Q|}$$

After testing question and answer generation module, MAP comes out to be 0.42.

5.3 Results for image captioning

For Image captioning we have used InceptionV3 model which gives us BLEU score as 0.374 and the results of the same are shown below:

Figure 5.6: Result for image captioning

Figure 5.7: Result for image captioning

Figure 5.8: Result for image captioning

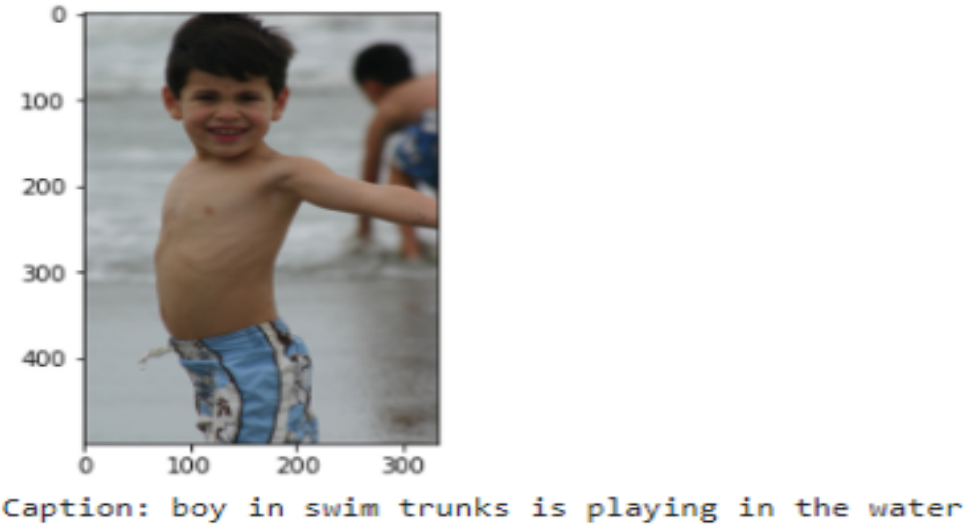
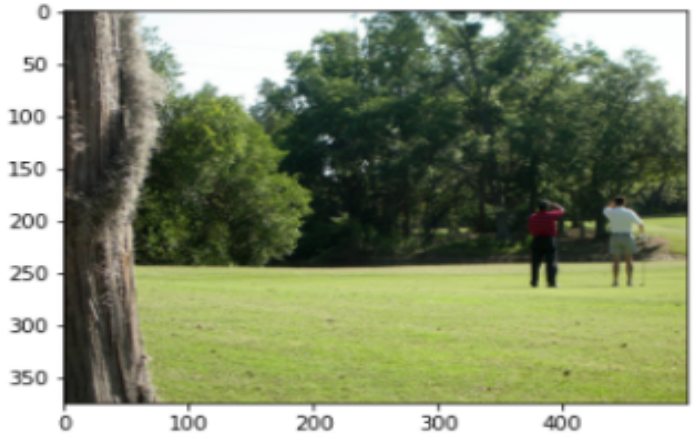


Figure 5.9: Result for image captioning

As shown in above figures(5.6 - 5.9) we can see that the captions are being generated accurately for the input pictures.

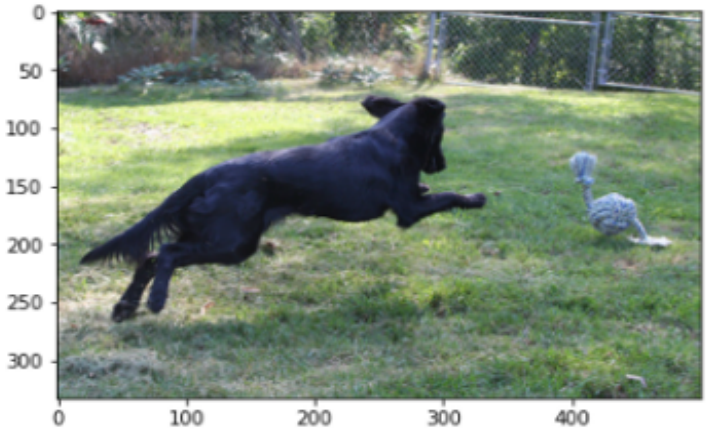
5.4 Results for question and answer generation

For question and answer generation we have used T5 based multitask qa-qg model which generates accurate questions and answers for the generated caption. The obtained results for the generated question and answers are given below.



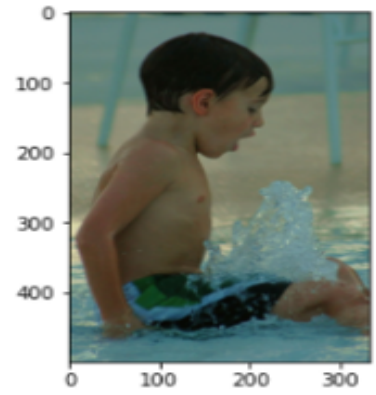

```
Caption: two people are walking through park
[{'answer': 'two', 'question': 'How many people walk through park?'}]
```

Figure 5.10: Result for question and answer generation



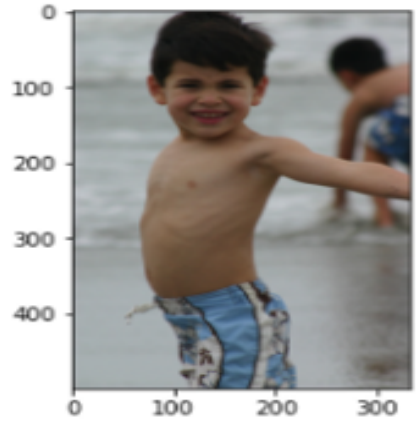

```
Caption: black dog is running through the grass
[{'answer': 'black dog', 'question': 'What is running through the grass?'}]
```

Figure 5.11: Suitable Caption




```
Caption: boy in swimming trunks is splashing in the water
[{'answer': 'boy', 'question': 'Who is splashing in the water?'},
 {'answer': 'swimming trunks',
  'question': 'What is the name of a boy who splashes in the water?'}]
```

Figure 5.12: Result for question and answer generation




```
Caption: boy in swim trunks is playing in the water
[{'answer': 'boy', 'question': 'Who is playing in the water?'}]
```

Figure 5.13: Result for question and answer generation

As shown in above figures(5.10 - 5.14) we can see that the questions and answers are being generated from the generated captions.

Chapter 6

CONCLUSION AND FUTURE SCOPE

We have proposed a framework for visual question answering model. A Flicker 8k dataset containing 8092 images of which 6000 images are used for training,1000 images for testing and 1000 images for development. An InceptionV3 model is being used in the proposed architecture to obtain better results for image captioning which when compared to other CNN models i.e. MobileNetV2 and VGG16 gave better accuracy and was robust than other models.For processing the language part,we used Show and Tell model which intakes image as an input and converts it to word vector which in turn is translated to caption using LSTM network(acts as decoder).For question and answering task we used fine tuned multitask qa qg model where the generated captions from image captioning module are fed.Then answers are generated from the captions and for question generation both the caption and the answers are being fed to the model which will give us both question and answer.

The model can be further fine tuned to improve the accuracy and for better captioning of images.Fine-tuned GRUs or BERT can be used for better question and answer generation.

REFERENCES

[1]Zhang, Shijie, et al. "Automatic generation of grounded visual questions."*arXiv preprint arXiv:1612.06530.*, 2016.

[2]Wu, Jialin, Zeyuan Hu, and Raymond J. Mooney. "Generating question relevant captions to aid visual question answering."*arXiv preprint arXiv:1906.00513*, 2019.

[3]Mora, Wu, Jialin, Zeyuan Hu, Issey Masuda, Santiago Pascual de la Puente, and X. Giro-I. Nieto."Towards automatic generation of question answer pairs from images."*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops.*, 2016.

[4]Lau, Jason J., Soumya Gayen, Asma Ben Abacha, and Dina Demner-Fushman. "A dataset of clinically generated visual questions and answers about radiology images."*." Scientific data 5.* ,no. 1 (2018): 1-10.

[5]Mostafazadeh, Nasrin, et al. "Generating natural questions about an image."*arXiv preprint arXiv:1603.06059.*, 2016.