Automated Image Caption Generator in Bangla Using Multimodal Learning

by

Mashiat Hasin Rodoshi
ID: 19201089
Moin Uddin Ahmed
ID: 19301095
Md. Sobhan Ashraf
ID: 19301046
Md. Galib Hasan Mim
ID: 19301094
Ashfia Khanam
ID: 18301231

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
January 2023

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Mashiat Hasin Rodoshi

Mashiat Hasin Rodoshi

ID: 19201089

Moin Uddin Ahmed

Moin Uddin Ahmed

ID: 19301095

Md. Sobhan Ashraf

Md. Sobhan Ashraf

ID: 19301046

galib Hasan

Md. Galib Hasan Mim

ID: 19301094

Ashfia khanam

Ashfia Khanam

ID: 18301231

Approval

The thesis/project titled "Automated Image Caption Generator in Bangla Using Multimodal Learning" submitted by

1. Mashiat Hasin Rodoshi (19201089)
2. Moin Uddin Ahmed (19301095)
3. Md. Sobhan Ashraf (19301046)
4. Md. Galib Hasan Mim (19301094)
5. Ashfia Khanam (18301231)

of Fall, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 19, 2023.

Examining Committee:

Supervisor:
(Member)

Farig Yousuf Sadeque

Dr. Farig Yousuf Sadeque
Assistant Professor
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Experiencing an image on-screen is a privilege that we often seem not to care about. A visually impaired person does not have that luxury. A system that can automatically produce closed captions of an image can thus help visually impaired people experience what's appearing on a digital screen. Research in this area has been in the forefront of multimodal machine learning for quite some time; but while a plethora of languages has benefited from all that research, Bangla has been left behind. For our thesis, we would like to build a Bangla Caption Generator using multimodal learning with high accuracy which automatically produces closed captioning in Bangla for digital images. The generator will be able to identify different objects in the image, relations among the objects and the actions happening in the image using neural networks. Combining the information collected, it may construct an information-rich, descriptive caption for the image. These captions can be later read aloud so that visually impaired people can get an idea about what is happening around them. This thesis aims to achieve further improvement upon the existing image caption generator in Bangla so that it can greatly help to improve the lives of visually impaired people as well as advance this research towards the state of the art. We have used the Flickr8k and Flickr30k datasets containing 8091 and 31783 images respectively and there are five Bangla captions for each image. We have used the VGG16, VGG19, ResNet50, InceptionV3 and EfficientNetB3 CNN architectures for feature extraction. Our best model has achieved a BLEU-1, BLEU-2, BLEU-3 and BLEU-4 score of 0.553197, 0.341976, 0.234436 and 0.113089 respectively.

Keywords: Image captioning; Machine Learning; CNN; LSTM; RNN; Deep Learning; Bangla; Natural Language Processing

Acknowledgement

Firstly, all praise to the Great Allah for whom our thesis has been completed without any major interruption.

Secondly, to our supervisor Dr. Farig Yousuf Sadeque for his kind support and advice in our work. He helped us whenever we needed help.

And finally to our parents. Without their throughout sup-port it may not be possible. With their kind support and prayer we are now on the verge of our graduation.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Despite the fact that Bangla is the world's seventh most spoken language by the total number of speakers [15], we are still unable to narrate any scenario or simply any image in Bangla, which is extremely difficult for visually impaired people, and people in rural areas who do not understand English. There are just a few options that generate suitable Bangla captions due to a lack of research that depicts the Bengali cultural exact features for photos. Although the solution is not simple to comprehend, the research requires a large amount of data containing native objects and captions. The Bangla caption generator is incomplete without such data. For example, if a photo depicts a woman wearing a maxi (a traditional Bangladeshi women's clothing), the system may misidentify the subject and display inaccurate results, such as a gown. To address the issue, we plan to develop a proper and precise Bengali caption generator that will assist all Bangla speakers in comprehending the scenarios and images they are now viewing or watching on a device.

Because of the recent increase in internet availability in Bangladesh, evaluating the image displayed on the internet has become crucial. There are normally English captions for all of the images displayed, and only a few other languages are available. However, Bangla captions are infrequent. As a result, all the Bengali people surfing the internet will tremendously benefit from our research.

Google originally introduced image caption generation in November 2014 [4], which translated photos into words. It is then matured with the use of data uploaded to the internet as well as the researchers' hard work. Nowadays, the English caption generator is extremely accurate, displaying the actual outcome of the image. However, there is very little research on Bengali image captions. One of the main goals of this study is to improve the production of image captions in our language.

1.1 Research Problem

Describing an image accurately in natural language is a task that has garnered a lot of interest due to its complexity. Image captioning combines both Computer Vision and Natural Language Processing (NLP). Here, we not only have to identify objects present in the image but also describe them in a way that seems logical and natural to a human. So it is a challenge to ensure that we can bring out the best in both fields. Image captioning has a lot of uses but the most prominent one is that it can make the lives of visually impaired people much easier. Colorblind people can know the actual color of an object while people with poor or no vision can know what an image looks like if the system is paired with a text-to-speech service. Image captioning can be paired with more advanced technology to provide live descriptions based on camera images or even count objects. A lot of extensive research has been conducted on image captioning for many languages but the number has not been as high for Bangla. This is a reason why we would like to carry out our research to build a Bangla Image Caption Generator using multimodal learning. The development of a Bangla Image Caption Generator using multimodal learning would radically alter the way we receive and perceive captions. This approach has the potential to be the successor to earlier attempts to caption images. It will be able to deliver more accurate captions as a result of its methodologies that combine computational power and knowledge of complex linguistic ideas. Not only will blind people benefit from this technology, but individuals, corporations, and the government will get a better understanding of the scenes being seen or investigated.

1.2 Research Objectives

We have some objectives that we would like to fulfill or explore while conducting our research. These are as follows: Applying active learning to the system if possible.

- To evaluate how well different image datasets may be used to generate image captions in Bangla automatically and to report our findings.
- To evaluate the effectiveness of a suggested CNN feature extraction model in terms of its ability to correctly identify features that are important for the precise generation of descriptive captions in Bangla.
- To explore the correctness of the generated descriptions and to establish the level of depth that is attainable when utilizing a Bangla picture caption generator with a variety of different image datasets.
- To investigate several approaches and architectural frameworks for merging different models to produce a full Bangla picture caption generator pipeline that is capable of producing captions with high levels of accuracy and language fluency that is comparable to that of a human user.
- To make the generated captions more accurate and useful for end-users by employing Machine Learning techniques such as Natural Language Processing (NLP) methodologies.
- To develop novel ways for evaluating the effectiveness of a Bangla image caption generator on a variety of real-world tasks, such as describing photos from a variety of domains or recognizing specific objects in photographs that will be used for captions.
- To analyse the various ways in which multimodal learning can be used to improve the accuracy of generated captions, even when there is a limited amount of data available.
- To explore further advanced machine learning methods such as Long Short-Term Memory (LSTM) networks to improve the quality of captions generated by our image caption generator.

1.3 Thesis Structure

There are six chapters in the thesis. In the first chapter, ”Introduction” an overview of the research problem and how it fits into the world is given. In the second chapter, ”Related Work” a review of the research problem-related literature is given. In the third chapter, ”Methodology” the ways that the research problem was looked into are explained. The results of the investigation are shown and talked about in the fourth chapter, ”Implementation and Results”. The fifth chapter, ”Challenges” talks about problems that might come up during the research process and how we tried to deal with them. In the last chapter, ”Conclusion,” the main results of the study are summed up.

Chapter 2

Related Work

2.1 Show and Tell: A Neural Image Caption Generator

In the paper “Show and Tell: A Neural Image Caption Generator”, a model is introduced which is based on a deep recurrent architecture that can be used to generate natural sentences that describe images [4]. The goal of this model was to maximize the probability of generating a sequence of words that would describe a given image with the highest accuracy from a predefined dictionary. The model is inspired by the latest achievements of sequence generation in machine translation. It is based on an end-to-end neural network that comprises CNN (Convolutional Neural Network) for feature extraction and LSTM (Long-Short Term Memory) for sentence generation. The image is mapped using CNN while the words are mapped using an embed model in the same space. The LSTM model is trained to predict every next word in a sentence based on the image it has seen and the previous words. Sentence generation is initiated with a special start word and ended by a special stop word. The model can generate sentences using two methods- sampling and beam search. The experiments carried out were conducted using a beam search method. The main challenge faced when training the models was overfitting. This problem was fixed by initializing the weights of the CNN component as a pertained model and this helped in generalization. Transfer learning showed both improvement and deterioration in accuracy depending on the dataset. The model generates descriptions that are both high quality and diverse, even descriptions which were not part of the training set. Experiments were carried out using PASCAL, Flickr30K and MS COCO datasets. The latest experiment using the COCO dataset rendered a BLEU-4 score of 27.7 which is considered the state-of-the-art [4].

2.2 Image Captioning with Semantic Attention

There are two image captioning approaches in practice which are top-down where the core of an image is converted into words and bottom-up where words are generated to describe various features of the image and then combined them together[5]. The paper “Image Captioning with Semantic Attention” proposes using these both approaches through a semantic attention model to generate image descriptions. In terms of image captioning, it described semantic attention as the ability to deliver a complete and coherent description of semantically imported things at the precise time when they are required [5]. It uses CNN to detect concepts that could be brought under attention in the bottom-up approach along with extracting top-down visual features to determine when and where attention should be used. The model combines the visual features and visual concepts through RNN (Recurrent Neural Network) to construct the caption of a image. This model can use concepts that are present at any resolution in the image and those that are not even visually present.In this paper, two models were used for visual attribution prediction. Te first one is utilizing a ranking loss as objective function to create a multi-label classifier and the other one is using a Fully Convolutional Network (FCN) to learn characteristics from local patches. Experiments were run on Flickr8k and MS COCO datasets. Using the attention model with FCN for visual attribute extraction, they achieved a better score than even the state-of-the-art approaches with the BLEU-4 scores 0.230 and 0.304 for both datasets respectively[5].

2.3 TextMage: The Automated Bangla Caption Generator Based On Deep Learning

This study ”TextMage: The Automated Bangla Caption Generator Based On Deep Learning” [10] proposed an automated image caption generation system in Bangla which is highly influenced by the architecture of “Show and Tell: A Neural Image Caption Generator” [4]. It combined the CNN, RNN and LSTM model to produce a caption where it follows the traditional procedure to extract images using CNN while generating languages through a RNN model consisting of LSTM cells. It used a tailored dataset for the proposed model named as “BanglaLekhaImageCaptions’ which is based on native South Asian geo-context. The dataset contains 9154 images along two human annotations in Bangla for each image. The paper produced an accuracy of 0.758565 for the training period while 0.643476 for the validation period in the CNN part. By implementing the Adam Streamlining method, it achieved RNN accuracy of 0.807854 during the training and after that, it merged the two models to prepare the complete framework and reached 91.65 accuracy in training while 73.97 in the validation.

2.4 Bangla Image Caption Generation through CNN-Transformer based Encoder-Decoder Network

In the paper “Bangla Image Caption Generation through CNN-Transformer based Encoder-Decoder Network” [12] which focuses on Self-attention, the backbone of Extended Neural models that are all used as the fundamental building blocks for Deep Convolutional Neural Networks. Image captioning is a function in natural language used to describe visual data by applying algorithms that interpret and represent connections underlying visual and textual information in order to produce a text output. The majority of research has been on encoder-decoding framework akin to the sequence-to-sequence paradigm. The architecture of merge or mixing, which is predominantly used for linguistic encoding CPTR, that substitutes CNN with a completely convolutional-free transformer encoder throughout the encoder section and explore a new sequence-to-sequence approach to the subtitling issue. The statistics are then combined into a single multi-modal layer that predicts the title term by a phrase. The MSCOCO team has recently released a new operating system for evaluating image subscribing systems. The project discovered that Deep sequence model approaches have delivered some astounding outcomes. In the field of Image subscripting, a search-based method was widely used which indicates the geographical relationship between quality and annotation. This Encoder decoder research employs the ResNet-101 algorithm model where the encoder is a representation in abstract form that compares all information learned by the Encoder Layer. Here, the encoder, containing two sub-modules: multi-headed attention and linked networks, performs the following operations: Feature Extraction, Positional Encoding, Multi-headed Attention, Dot product of Query and Key, etc. The decoder then integrates all of the provided data. The decoder consists of Nd identical layers, with each layer requiring its own sub-layer to function. In terms of quantitative outcomes, the model set a new standard with scores of 0.694 on BLEU-1, 0.630 on BLEU-2, 0.582 on BLEU-3, and 0.337 on METEOR.

2.5 Improved Bengali Image Captioning via deep convolutional neural network-based encoder-decoder model

This paper [11] introduces a Improved Bengali Image Captioning (BIC) model which is an encoder-decoder system, based on a deep convolutional neural network. This model can be used to improve image search, provide a description of CCTV footage, and more. This model encodes sequence information using a one-dimensional convolutional neural network (CNN) and an image encoder that was pre-trained using a ResNet-50 model. CNNs are used in the process of extracting image features, while LSTMs are used in the process of decoding captions. The generator of the caption first determines the probability of each vocabulary word, and then, using a greedy algorithm, it selects the most appropriate word. Five different metrics, including BLEU-1, CIDEr, METEOR, ROUGE, and SPICE, were utilized in order to assess the performance of this model. According to the findings, this model received an overall score of 0.651 for the BLEU-1 test, 0.572 for the CIDEr test, 0.297 for the METEOR test, 0.434 for the ROUGE test, and 0.357 for the SPICE test[11]. This demonstrates that this model can be utilized effectively to generate accurate captions for Bengali images while maintaining a high level of accuracy and efficiency. [11]

2.6 Chittron: An Automatic Bangla Image Captioning System

This study [7] covers the construction of a Deep Neural Network-based automatic Bangla image caption generator. It utilized a dataset titled BanglaLekha-ImageCaptions including approximately 16000 photos and a single, excessively descriptive annotation for each image. In the suggested model, the process of captioning was divided into two sections: collecting relevant image features and then generating a linguistic description based on the extracted features. This report's pre-trained image model extracts image embeddings using existing models such as im2txt, NeuralTalk, etc., and then predicts single word at a time from the LSTM layers. In order to pre-train the image model, the VGG16 model was employed with slight modification. It is trained using the back propagation method from beginning to end. Using Basic Stochastic Gradient Descent, it minimize the category cross entropy of the output of the stacked LSTM layers. The model obtained a BLEU score of only 2.5 because each image in the dataset had only one caption. However, the approach mainly emphasized on better qualitative outcomes.

2.7 Very Deep Convolutional Networks for Large-Scale Image Recognition

The authors of this paper tested the convolution network depth accuracy on large scale image recognition [1]. Experiments have been conducted with architecture with various depths starting from 11 to 19 weight layers. Smaller convolution filters have been used in the architecture. 1*1 convolutions showed that linear transformation of input channels and non-linearity afterwards increases the discriminative capability of the decision function. This system also works on varying image scales. During training, the image gets rescaled to set the length of the shortest side to S. Then, the image gets corp 224*224. In multi scale, S is sampled randomly from[256,512]. This can be seen as a type of data augmentation by scaling jittering. Here, a single model is trained to recognize a wide range of scales of objects. In dense evaluation, the fully connected layers are transformed to convolutional layers at testing time. Uncropped images are passed through the convolutional net in order to get a dense class score. The scores are averaged to obtain the final fixed-width class posterior. This is compared against multiple corps of the test image score passing through CNN. Multi-corp evaluation is slightly better than dense evaluation. One of the strong sides of this paper is the very thoughtful design of architecture. They have conducted experiments to study effect of depth, LRN, 1*1 convolution, pre-initialization of weights, image scale etc. One of the drawbacks of the architecture is not getting finite time details of how much time it is required to train.

2.8 Rethinking the Inception Architecture for Computer Vision

In the paper, it has been made clear that improvements in deep convolutional architecture can be used to improve the accuracy of most computer vision tasks because they rely on high quality image features [3]. Dimension reduction is greatly useful for getting accurate results.The paper has explored different ways of factorizing convolutions to improve computation efficiency. With proper factorization, faster training can be possible due to disentangles training. Replacing nxn convolutions by 1xn and later nx1 convolutions saves more computation cost. However, the paper found that this does not work well for all layers except in medium-grid sizes of mxm where 20>=m>=12. In this level, one can use 1x7 convolutions followed by

7x1. Three 3x3 convolutions have been used. Modifying the InceptionV2 model, in the inception part, there are 17x17 grid with 768 filters that have been transformed using a grid reduction method. The model was trained for 100 epochs with batch size 32. The RMSProp optimizer and a learning rate of 0.045 have been used. The best version of the InceptionV3 model reached 21.2% top-1 and 5.6% top-5 error for a single corp evaluation. The ensemble model reached 3.5% top-5 error with multi-corp evaluation.

2.9 BanglaNLG: Benchmarks and Resources for Evaluating Low-Resource Natural Language Generation in Bangla

BanglaNLG is a comprehensive benchmark for evaluating natural language generation (NLG) models in Bangla, a widely spoken yet low-resource language in the web domain [13]. This model aggregates three challenging conditional text generation tasks under the BanglaNLG benchmark, such as summarization, question answering and dialogue generation. To further improve the performance of NLG models in Bangla, a clean corpus of 27.5 GB of Bangla data was used to pretrain BanglaT5, a sequence-to-sequence Transformer model for Bangla. The results show that BanglaT5 outperforms mT5 (base) by up to 5.4% in all of these tasks, demonstrating its effectiveness as a state-of-the-art NLG model for Bangla.

2.10 An Updated Evaluation of Google Translate Accuracy

Since 2011, Google Translate has seen an improvement of 34% in BLEU scores after being reevaluated using the same test text as the older version of this paper [6]. In 2016, the service was updated to use a Neural Machine Translation model. As a result, its machine translation improved from 3.694 to 4.263 which is almost near human level in quality. Google has reported that their service has made a 69% improvement in accuracy by using six language pairs for testing.

2.11 EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks

Efficient net is a convolutional neural network architecture whose main goal was to make models better in terms of both accuracy and speed [8]. The main goal of the architecture is to make scaling up CNNs easy and effective. By studying model scaling and figuring out that network depth, width, and image resolution can be carefully managed, it may be possible to improve performance. Putting forward a new method for scaling uniformly. Using a program called EfficientNets to search neural networks and design a new baseline network. ConvNet is 8.4 times smaller and 6.1 times faster. But it is hard to figure out how to make ConvNet bigger. The compound scaling method is an important part of EfficientNet. By changing the network's depth, width, and resolution at the same time. More layers are added to the network to make it deeper, more channels are added to each layer to make it wider, and larger input images are used to improve the network's resolution. The scaling process can also be done with ConvNet, but it has to be done randomly, which is a problem. The process of arbitrary scaling requires tedious manual tuning, but the results are still not the best. First of all, they noticed that scaling up any dimension improves accuracy, but the improvement gets smaller as the model gets bigger. Second, if we want ConvNet scaling to be more accurate and efficient, we must make sure that all the network's dimensions are in balance. The paper shows that it is important to keep all the areas in balance, which can be done by scaling each by a constant ratio. Using a set of fixed scaling coefficients, the method scales the width, depth, and resolution of the network in the same way. Using a multi-objective neural architecture search, the baseline is made. Overall, EfficientNet is a CNN architecture with state-of-the-art performance because it is very efficient and effective. The paper describes a method for scaling up baseline ConvNet that is very effective and easy to use. This shows that an EfficientNet with a mobile size can scale up a lot and work better.

2.12 Deep Residual Learning for Image Recognition

ResNet, or Residual Network, is regarded as one of the most significant contribution in deep learning and computer vision research. Deep neural networks have always encountered two major obstacles: vanishing gradient and degradation problem. As weights in neural models are fixed in the final layers by comparing the original value to the predicted result, the weights in the earlier layers converge to such low level that it makes those layers' learning almost negligible in deeper models, a phenomenon known as the vanishing gradient problem. With the inclusion of the ReLU activation function and renormalization, the problem of vanishing gradients was resolved. Afterwards, it is thought that stacking additional CNN layers may improve accuracy. Hence, subsequent models have become deeper than their predecessors. In 2012, AlexNet began using only five CNN layers, whereas VGGNet and Inception followed with 16, 19, and 22 layers, respectively. However, the deeper models began to encounter another issue. As model depth increases, the accuracy begins to saturate and then rapidly declines. In the following paper [2] the authors claim that the issue is not due to overfitting and demonstrate that adding more layers to an appropriate deep learning model increases training error. This is referred to as the degradation problem. To gain a better understanding, imagine two models, one shallower with x layers and the other, its deeper counterpart with y layers, where y is more than x. It is anticipated that the accuracy of the deeper network will be significantly greater or at least equivalent to that of the shallower network. In practice, however, it is observed that the accuracy begins to deteriorate rather than improve. The author hypothesized a solution in which the deeper layers propagate the information directly from the shallower layers in order to address this issue. The initial x layers of the deeper model are identical to those of the shallower model, while the subsequent (y-x) levels are simply identity mapping. They showed that other solutions available at the time were not better than what they had suggested.

Consequently, the authors presented a novel solution: a deep residual learning architecture with skip connection or shortcut connections. To better comprehend residual learning, consider F(x), a function learned by the stack of layers. Considering an additional layer after the skip connection H(x) = F(x) + x. Here, x represents the initial value which comes by skip connection or a shortcut, while F(x) represents the residue or changes to the value. Regardless of what the value F(x) learned is, it is the residue value and it is used to change the initial value. Hence, the term Residual Learning was introduced. To make H(x) the identity function, F(x) or the residue value has to be zero, which is fairly simple to learn compared to the complicated modification of weight and bias values required by earlier models, thus resolving the degradation issue. In contrast to the other models, which exhibited an increase in training error with increasing depth, this study demonstrated that it is extremely simple to optimize even with 1000 layers. In addition, it enhances accuracy by adding more layers without degrading like previous models.

Chapter 3

Methodology

This thesis' purpose is to explore the methods of existing image caption generators and provide an image caption generation model that can improve upon their accuracy.
The workflow of the entire process has been summarized below:

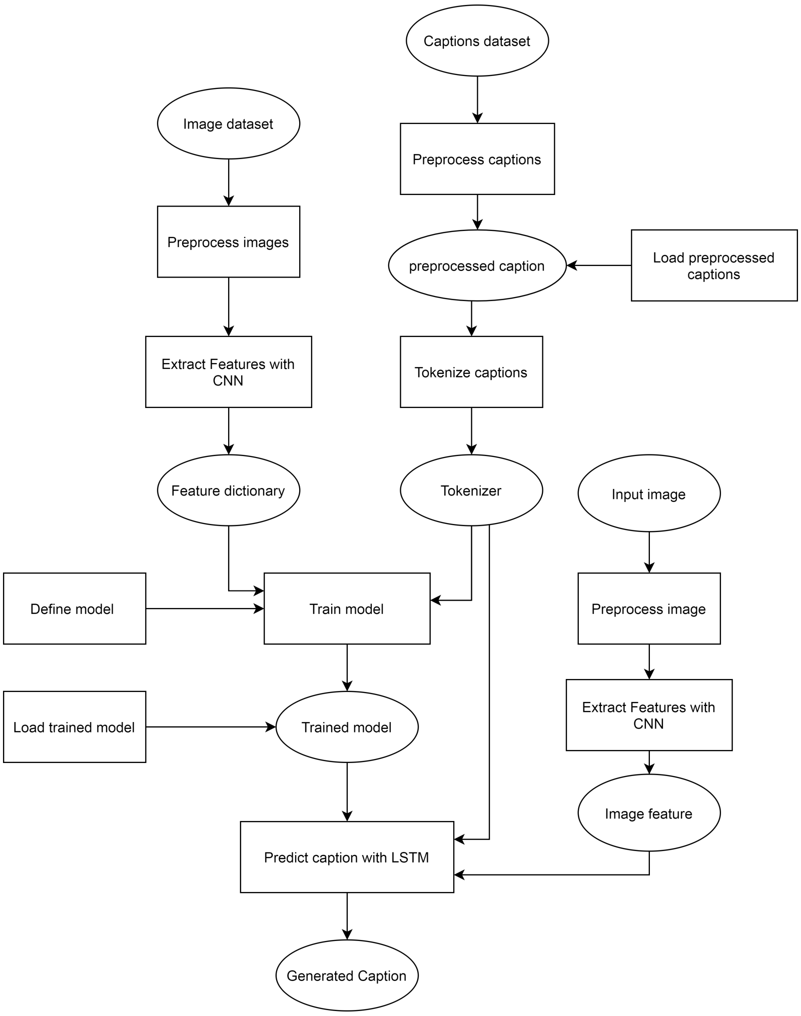


Figure 3.1: Flow Chart of workflow of proposed Bangla image caption generator

The use of CNN for image processing and RNN with LSTM for Bangla caption generating are the primary characteristics of our system.

The entire procedure can be split into two steps:

1. Feature Extraction: After importing an image, it is required to identify its overall characteristics and objects in order to determine which features corresponds to which scenario. A CNN architecture is used to identify the characteristics.

2. Generating the Bangla captions: We gain a vector representation of the features after feature extraction. Then the features are matched with words from our preprocessed captions using an RNN approach consisting of LSTM. Finally, we get a generated Bangla caption for the entire image.

3.1 Input Data

Working with a good dataset is a crucial part of any kind of machine learning project. It plays a very important role in determining the accuracy of the model used. The performance of a machine learning (ML) model depends greatly on the quality of the data [9]. A good dataset allows the model to acquire more relevant information which helps to provide better predictions. So, for our models, we have decided to use a dataset that is both rich in terms of quality and quantity.

We have settled for the “Flickr8k” image dataset which is the new benchmark collection of image based descriptions, consisting of 8091 images [18]. This dataset contains images containing a wide variety of situations and actions. For image captions, we are using the “BAN-Cap: English-Bangla Image Descriptions Dataset” which is a collection of captions of images from the Flickr8k dataset that have been annotated by qualified contributors [14]. There are five captions for each image narrating the image in unique ways which can be useful to provide diversity to the model.

We have also decided to use the ”Flickr30k” image dataset which is an extension to the Flick8k dataset [17]. It has 31783 images and 5 annotations for each image. The available captions are in English and it is a time consuming process to annotate every image in Bangla from scratch or to translate the English captions to Bangla manually. Therefore, we have decided to translate the available captions to Bangla using machine translation. We have tried two machine translation services which are Google translate and BanglaNLG. The image captioning models have been trained using these translated captions.

	Flickr8k	Flickr30k (Google translate)	Flickr30k (BanglaNLG)
Number of images	8091	31783	31783
Number of captions	40455	158916	158916
Vocabulary size	15666	26024	31783
Maximum caption length	32	66	122

Table 3.1: Dataset information

3.2 Data Pre-processing

3.2.1 Image feature extraction

A CNN (Convoluted Neural Network) architecture collects the main features from the dataset images. Usually a CNN architecture starts by identifying the edges of the current image. Then it starts to identify shapes and colour groups. The important features are passed onto the next layer. The features get more defined with each successive layer. The max pooling layers return the most relevant features in the activation map. This process continues till we reach the final layer. We acquire a feature vector with the prominent features in the image in this way.
We have tested our model using image features extracted by VGG16, VGG19, Resnet50, InceptionV3, EfficientNetB3. We have also used ensembles of VGG16 and in another instance, EfficientNetB3. This helps to improve the accuracy of the generated captions.

VGG16 and VGG19

VGG16 has 16 layers that have weights. It has 138 million parameters. It takes images of size 224x224 square pixels with 3 RGB channel as input. It extracts the photo features as a vector of 4096 elements. VGG19 has three more trainable layers than VGG19. The inputs are also same for this model.

ResNet50

ResNet50 has 50 trainable layers. It has more than 23 million parameters. Like VGG16, it also takes images of size 224x224 square pixels with 3 RGB channel as input. It extracts features in a vector of 1000 elements.
ResNet solves some of the limitations of VGGNet. Firstly, it has lesser parameters meaning it is more computation friendly. Secondly, it dolves the vanishing gradient problem of VGGNet where the derivative of backpropagation to initial layers become almost zero. ResNet uses shortcut connections between neurons to solve this vanishing gradient problem.

InceptionV3

InceptionV3 is 48 layers deep. It has under 25 million parameters and takes images of size 299x299 square pixels as input and gives a feature vector of 2048 elements.
InceptionV3 is the advancement of InceptionV1. The main concept of inceptionV1 was that the model learns from parallel filters of different sizes at same level. So objects of various sizes can be identified more effectively by the model. 1x1 convolution filters are added to bigger convolutions to reduce channels and thus reduce parameters. This way less computation resources are used.
InceptionV3 improves on the existing concepts of InceptionV1. One of it's main focuses is factorization to smaller convolutions. For example, 5x5 convolution filters is 25 parameters. But (3x3)+(3x3) is 18 parameters. So, we get less parameters if we use two 3x3 convolutions instead of one 5x5 filter. This was further explored to see that factorizing into assymetric convolutions was more effective in some cases. For example, 3x3 convolutions is 9 paramrters But (1x3)+(3x1) convolutions is 6 parameters.

InceptionNet is very computation friendly but because ofthat its accuracy may be compromised.

EfficientNetB3

EfficientNetB3 has 5 blocks, 17 layers, and a total of 28 million parameters. The input image size for EfficientNetB3 is 300x300 pixels. It gives feature vector of 1536 elements. It has an output stride of 32 and uses a depthwise separable convolution with a width multiplier of 1.2. The main idea behind EfficientNet increasing accuracy by model scaling. Width, depth and resolution are the three things that can be scaled up. Width scaling is increasing feature maps, depth scaling is increasing layers and resolution scaling is increasing input image size. EfficientNet uses a method called compound scaling to ensure that these three things are scaled up proportionally. It uses three parameters alpha, beta and gamma to find out at what rate to scale them up.

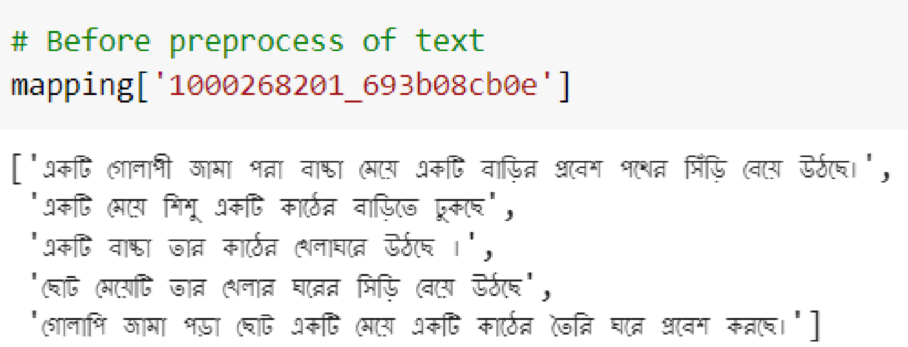
First we loaded the model to extract features from the images. The models are restructured into taking images as input and the output layer is without the final output layers. We have converted every image to the size supported by the architecture we used. Then we convert the image pixels to a numpy array and reshaped it. After this, we prepared the images for the architecture with the preprocess_input function. Finally, we predicted the features of the images and stored them as data in a dictionary with key being the image id.

3.2.2 Ensemble models

Ensemble models are a type of machine learning technique that combine the results of multiple individual models to produce results that are superior to those produced by any single model used on its own. This is accomplished by combining the estimates produced by a number of different models and then taking the mean of those forecasts in order to arrive at a more precise estimate. During the course of this research, we are developing numerous iterations of the VGG16 and EfficientNetB3 models and attempting to forecast their results. We are able to create an ensemble model of VGG16 and an ensemble model of EfficientNetB3 by averaging the outputs of these models. After getting their outputs, we are able to see that the meaningful result we are getting is superior to what any single model could produce on its own.

3.2.3 Caption pre-processing

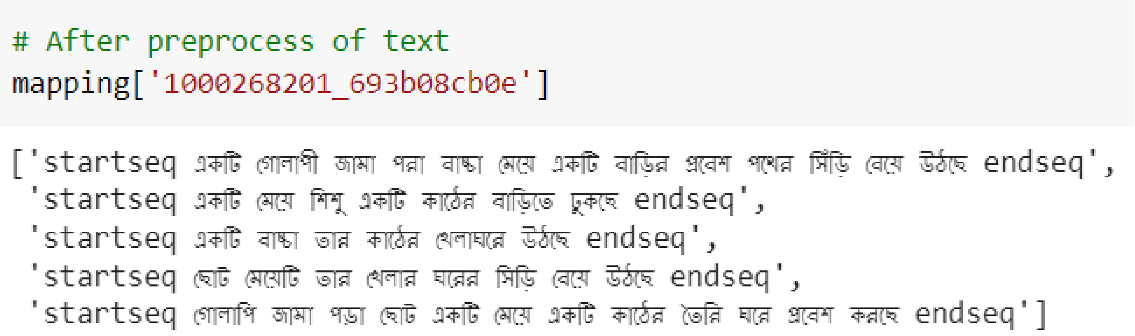
The caption dataset we have chosen comes in the form of a .csv file. It contains the image file name, English caption and Bangla caption. At first, we extract the image file names and the corresponding Bangla captions and save them in a separate descriptions.txt file. This ensures that the model can retrieve this data faster upon future executions from just loading the text file. Next, we collect the isolated captions from the descriptions file and map the captions to the image id in the form of a dictionary. Then the mapped captions are cleaned. We remove unnecessary information such as numbers, special characters and punctuation. ‘startseq’ and ‘endseq’ are added to the front and back of each caption for aiding in sequence creation later on.



```
# Before preprocess of text
mapping['1000268201_693b08cb0e']
```

```
['একটি গোলাপী জামা পরা বাচ্চা মেয়ে একটি বাড়ির প্রবেশ পথের সিঁড়ি বেয়ে উঠছে।',
 'একটি মেয়ে শিশু একটি কাঠের বাড়িতে ঢুকছে',
 'একটি বাচ্চা তার কাঠের খেলাঘরে উঠছে ।',
 'ছোট মেয়েটি তার খেলার ঘরের সিড়ি বেয়ে উঠছে',
 'গোলাপি জামা পড়া ছোট একটি মেয়ে একটি কাঠের তৈরি ঘরে প্রবেশ করছে।']
```

Figure 3.2: Captions before pre-processing



```
# After preprocess of text
mapping['1000268201_693b08cb0e']
```

```
['startseq একটি গোলাপী জামা পরা বাচ্চা মেয়ে একটি বাড়ির প্রবেশ পথের সিঁড়ি বেয়ে উঠছে endseq',
 'startseq একটি মেয়ে শিশু একটি কাঠের বাড়িতে ঢুকছে endseq',
 'startseq একটি বাচ্চা তার কাঠের খেলাঘরে উঠছে endseq',
 'startseq ছোট মেয়েটি তার খেলার ঘরের সিড়ি বেয়ে উঠছে endseq',
 'startseq গোলাপি জামা পড়া ছোট একটি মেয়ে একটি কাঠের তৈরি ঘরে প্রবেশ করছে endseq']
```

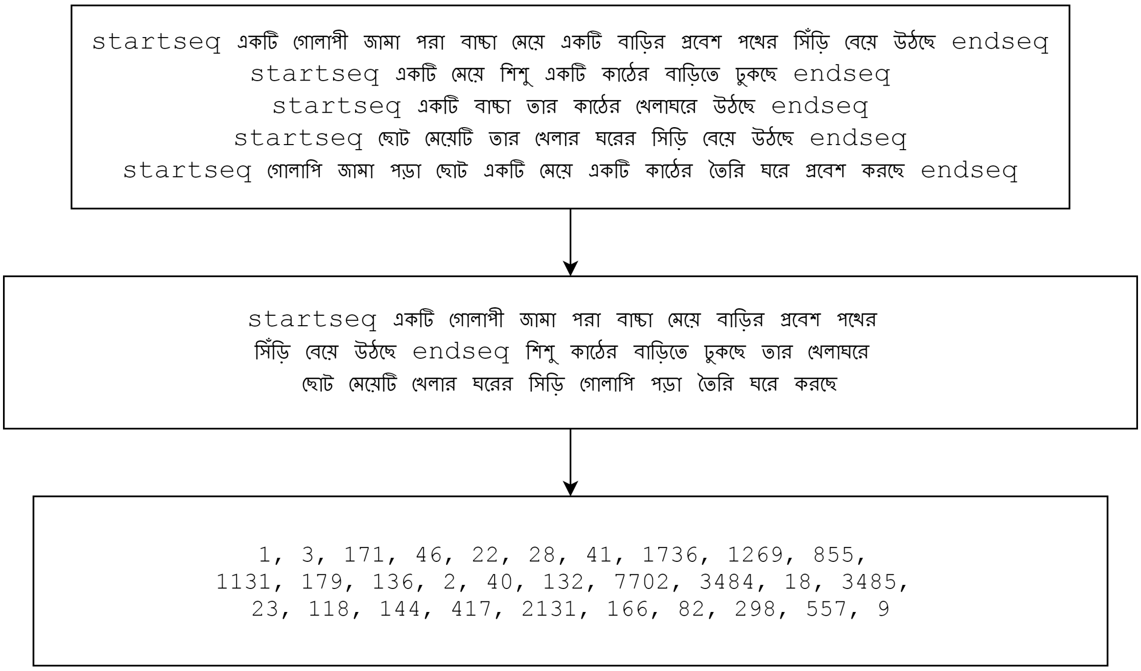
Figure 3.3: Captions after pre-processing

3.3 Tokenizing caption

The words in the descriptions need to be transformed into numbers before they can be used as input in the model. This is done by mapping every unique word to an integer value. We have done this by using the tokenizer class provided by keras.

3.4 Finding vocab size and max length

The vocab size is the total number of unique words in the caption dataset. So, just by getting the length of the tokenizer, we can get the vocab size. On the other land, max length is the maximum length of a given caption in the dataset.



startseq একটি গোলাপী জামা পরা বাচ্চা মেয়ে একটি বাড়ির প্রবেশ পথের সিঁড়ি বেয়ে উঠছে endseq
startseq একটি মেয়ে শিশু একটি কাঠের বাড়িতে ঢুকছে endseq
startseq একটি বাচ্চা তার কাঠের খেলাঘরে উঠছে endseq
startseq ছোট মেয়েটি তার খেলার ঘরের সিড়ি বেয়ে উঠছে endseq
startseq গোলাপি জামা পড়া ছোট একটি মেয়ে একটি কাঠের তৈরি ঘরে প্রবেশ করছে endseq

startseq একটি গোলাপী জামা পরা বাচ্চা মেয়ে বাড়ির প্রবেশ পথের
সিঁড়ি বেয়ে উঠছে endseq শিশু কাঠের বাড়িতে ঢুকছে তার খেলাঘরে
ছোট মেয়েটি খেলার ঘরের সিড়ি গোলাপি পড়া তৈরি ঘরে করছে

1, 3, 171, 46, 22, 28, 41, 1736, 1269, 855,
1131, 179, 136, 2, 40, 132, 7702, 3484, 18, 3485,
23, 118, 144, 417, 2131, 166, 82, 298, 557, 9

Figure 3.4: Tokenization

3.5 Train Test split

The dataset is split into two parts- training and testing. The training dataset is used to train the model and the testing dataset is used to test the model after training to see its performance. The training dataset for our model contains 90% of the total data available in the dataset.

3.6 Defining the model

The model we are using comprises an encoder model followed by a decoder model. The encoder model has two input layers which take the image feature and the sequence as input. The image feature shape depends on the CNN model used and it goes through a dropout layer of value 0.4 where noise and unnecessary details are removed to reduce overfitting to the training data. Then it goes through a dense layer of 256 neurons. The sequence contains the maximum length of words possible for the captions which is 34 for our Flick8k captions, 66 for the Flickr30k captions translated by google translate and 122 for the Flickr30k captions translated by BanglaNLG. This sequence goes through an embedding layer followed by a dropout layer. Then it goes through an LSTM layer with 256 memory units where the output sequence of words are generated.
Recurrent Neural Networks (RNNs) are a type of artificial neural network that are used to process sequential data. They are capable of learning patterns in data over time and can be used for a variety of tasks, such as language translation, speech recognition, and image captioning. RNNs have the ability to remember information from previous inputs and use it to inform future decisions. LSTM (Long Short-Term Memory) is a type of RNN (Recurrent Neural Network) architecture that can to store information for long periods of time. It can learn to improve its predictions

when more data is fed accessible to it. As a result, our Image captioning model has made use of them. LSTM networks are made up of memory cells that are connected to one another in a chain-like arrangement. Features extracted by a CNN and the input sequence of words are sent to the memory cells. The network generates a word based on the data that it stores in the memory cells as it processes the feature of the image and the input sequences. The output word is again sent to the memory cells along with the image to generate the next word. This procedure is carried out repeatedly until an accurate caption is generated.
Afterwards, the decoder model combines the outputs with an addition layer. Then this output is fed to a dense layer of 256 units followed by a final dense output layer that uses a softmax activation to predict the next word in the sequence for the entirety of the output vocabulary.

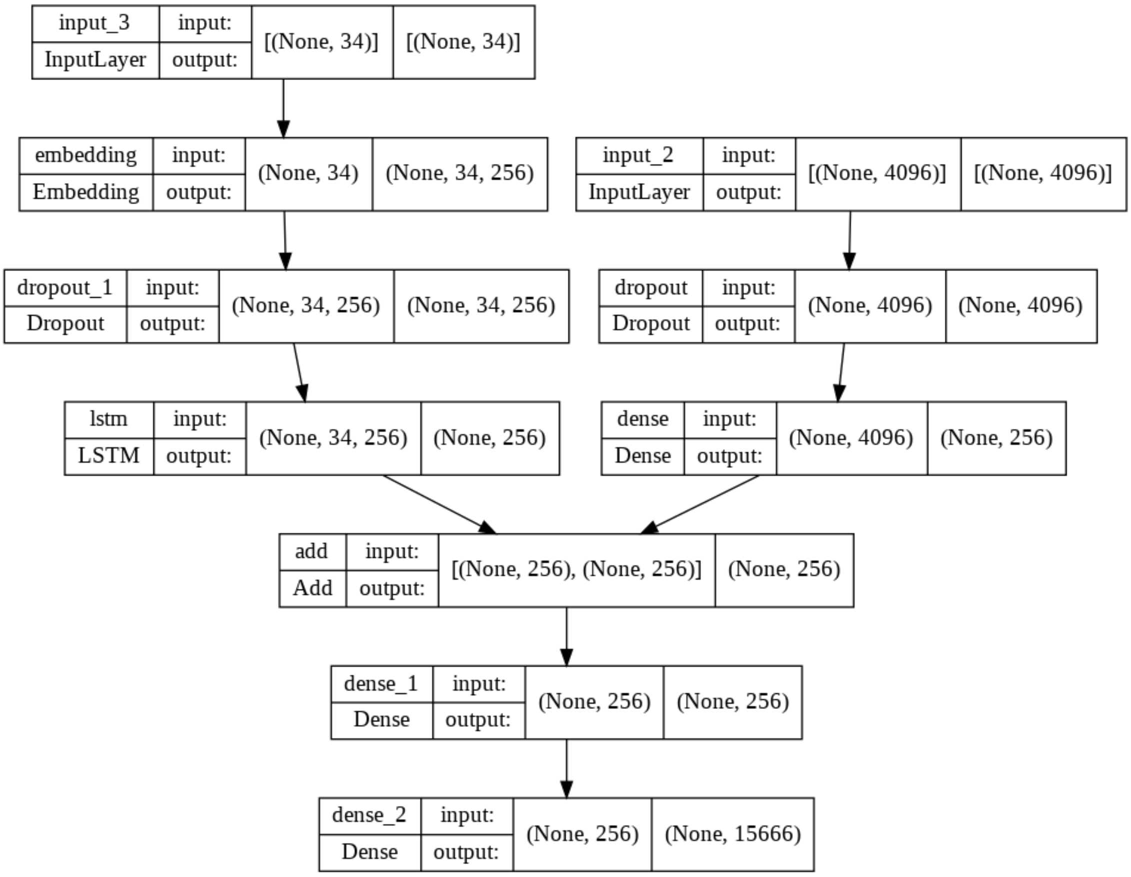

input_3 | InputLayer | input: | output: | [(None, 34)] | [(None, 34)]
embedding | Embedding | input: | output: | (None, 34) | (None, 34, 256)
input_2 | InputLayer | input: | output: | [(None, 4096)] | [(None, 4096)]
dropout_1 | Dropout | input: | output: | (None, 34, 256) | (None, 34, 256)
dropout | Dropout | input: | output: | (None, 4096) | (None, 4096)
lstm | LSTM | input: | output: | (None, 34, 256) | (None, 256)
dense | Dense | input: | output: | (None, 4096) | (None, 256)
add | Add | input: | output: | [(None, 256), (None, 256)] | (None, 256)
dense_1 | Dense | input: | output: | (None, 256) | (None, 256)
dense_2 | Dense | input: | output: | (None, 256) | (None, 15666)

Figure 3.5: Model structure

3.7 Model training

A set of data generated by a data generator is fed to the model to train it. It takes the features dictionary, captions dictionary, tokenizer, vocab size, batch size and max length of captions as input to generate input and output pairs for the model. For every image in the dataset, the data generator generates input sequences and output sequences for every caption for that image. The training is done recursively. First, the feature vector for the image and the first word in the caption in the form of a sequence are given as input to generate the next word of the caption. The output is validated against the actual output sequence. As we run more epochs, the loss reduces and the accuracy of the predictions increases. Afterwards, the image and the first two words are given as input to generate the third word. In this way, sequences of words are generated. For example:

x1	x2	y
photo	startseq	একটি
photo	startseq একটি	বাচ্চা
photo	startseq একটি বাচ্চা	তার
photo	startseq একটি বাচ্চা তার	কাঠের
photo	startseq একটি বাচ্চা তার কাঠের	খেলাঘরে
photo	startrseq একটি বাচ্চা তার কাঠের খেলাঘরে	উঠছে
photo	startseq একটি বাচ্চা তার কাঠের খেলাঘরে উঠছে	endseq

Table 3.2: Training the model with generated sequences

Here, the image and encoded text are given as input and the output is the predicted next word. In this example, there are 7 input and output pairs. The separated input and output pairs are stored in lists. This process continues up until the last word of the sentence has been inputted along with all the previous words for all captions for all images in the training dataset.

We have used 25 epochs to train our model. We have found this number of epoch to be ideal as lesser epochs were not ensuring good results and higher epochs were causing the model to overfit to the training dataset. Since we are dealing with a large amount of data, it has been fed to the model in batches. We have used 34 batches to train our data. So the entire dataset was trained in 214 steps.

We have collected the accuracy and loss per epoch to plot graphs to get the gist of the model's performance. These graphs can be seen in Chapter 4 of this report.

Chapter 4

Implementation and Results

4.1 Training Performance

The loss vs epoch and the accuracy vs epoch graphs using loss and accuracy during training is shown below-

4.1.1 Flickr8k

VGG16

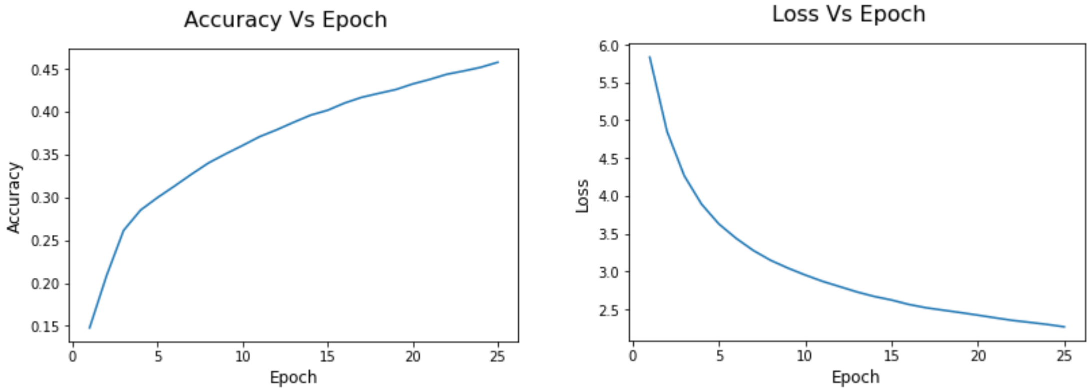


Figure 4.1: Flickr8k: Graphs for VGG16

VGG19

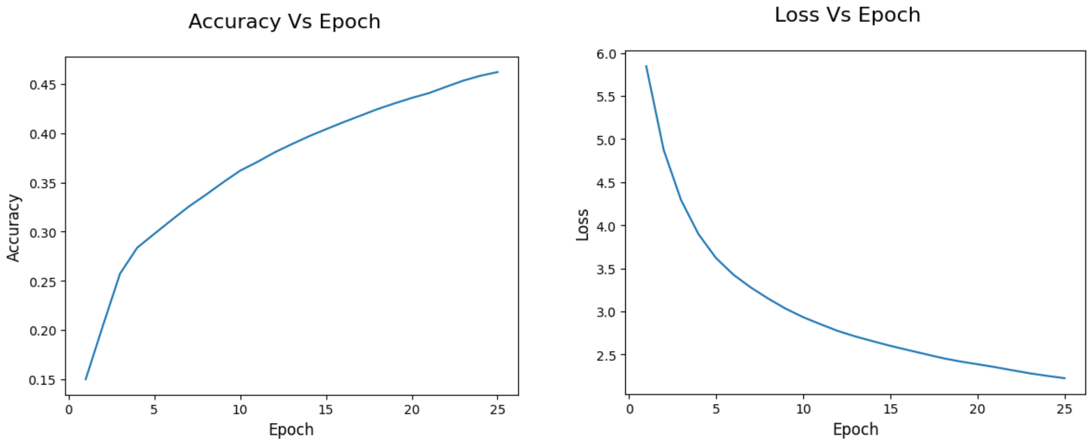


Figure 4.2: Flickr8k: Graphs for VGG19

ResNet50

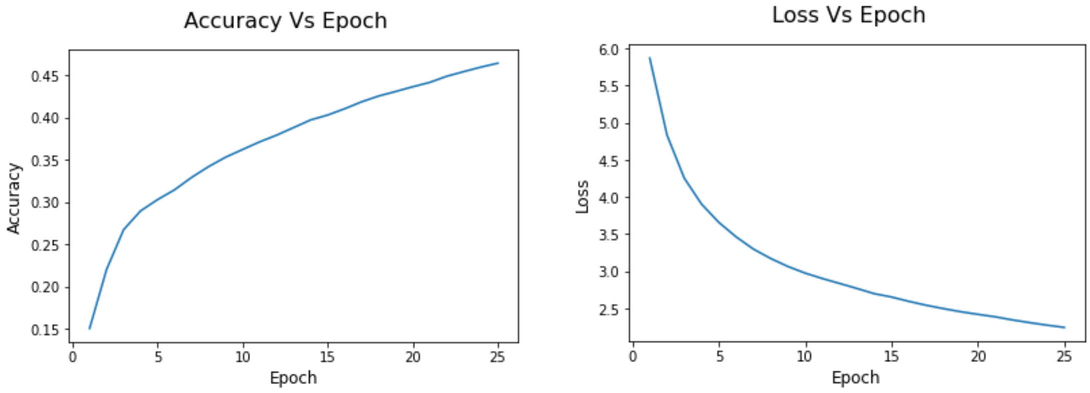


Figure 4.3: Flickr8k: Graphs for ResNet50

InceptionV3

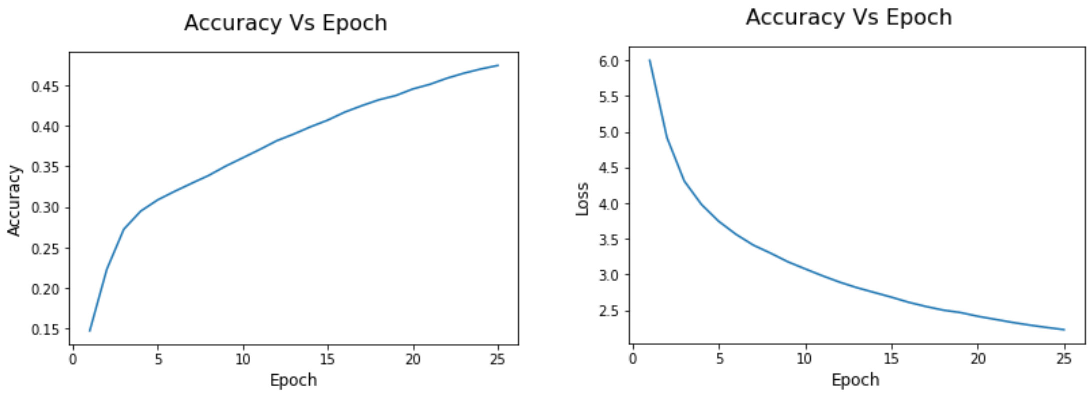


Figure 4.4: Flickr8k: Graphs for InceptionV3

EfficientNet

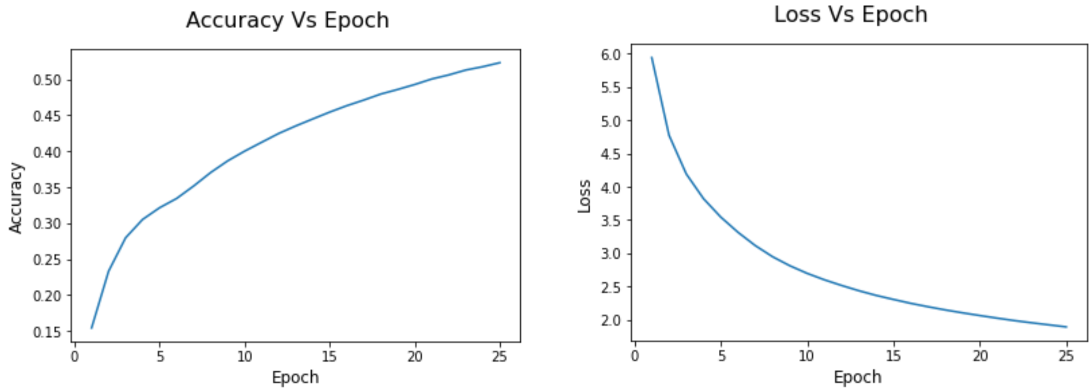


Figure 4.5: Flickr8k: Graphs for EfficientNet

4.1.2 Flickr30k

VGG16

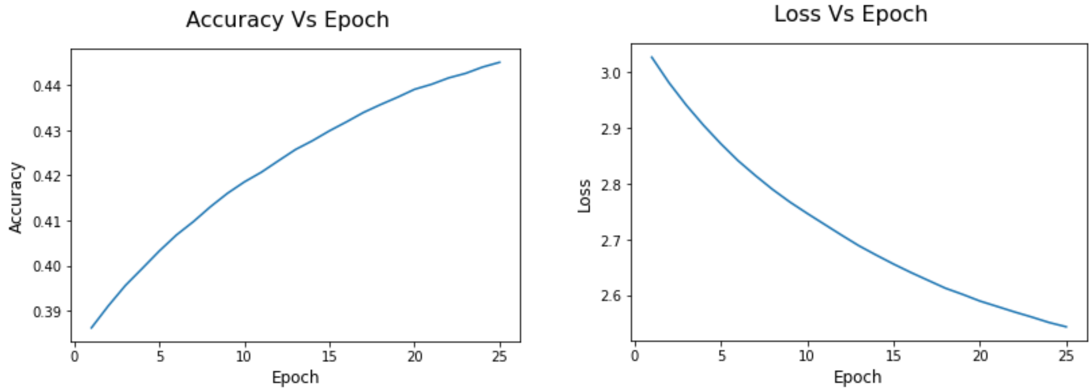


Figure 4.6: Flickr30k: Graph for VGG16

4.2 Caption Generation

To predict the caption, we need the model, feature vector of the image, tokenizer and max length of the caption. The string ’startseq’ indicates the start of the prediction. After this, we will iterate over the max length of sequence which is 34 for Flickr8k dataset, 66 for Flickr30k Google Translated captions and 122 for Flickr30k BanglaNLG captions. The model then predicts the next word. We find the index of the token with the highest probability. Then, that token is converted to word. If a word exists, it will get concatenated to the previous of generated words. Otherwise, the loop will break. If the word is ’endseq’ which marks the end of the caption, then the generated caption will be returned.

We have randomly selected some images and corresponding generated captions by the model with different CNN architecture. They are shown below:

4.2.1 Flickr8k

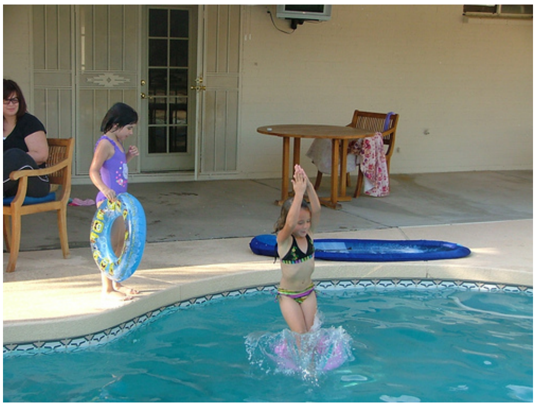


Figure 4.7: Flickr8k: Test image 1

Actual captions:
startseq একজন মহিলা দেখছেন দু'টি শিশু পুলে খেলছে endseq
startseq একটি যুবতী একটি পুলের মধ্যে ঝাঁপিয়ে পড়ছে, যখন অন্য একজন ইনারটিউবটির সাথে দাঁড়িয়ে আছে endseq
startseq একটি অল্প বয়সী মেয়ে প্রথমে একটি সুইমিং পুলে পা ছুঁড়ে যখন অন্য ছোট মেয়ে এবং মহিলা দেখছে endseq
startseq দু'জন ছোট মেয়ে তাদের পরিবারের পিছনের উঠোন সুইমিংপুলে খেলছেন যখন একজন মহিলা দেখছেন endseq
startseq দুটি ছোট মেয়ে একটি হোটেলের পুলের সাহায্যে খেলছে endseq

Predicted captions: VGG16: startseq একটি শিশু পুলে ঝাঁপিয়ে পড়ছে endseq
VGG19: startseq একটি ছেলে পুলে ঝাঁপিয়ে পড়ে endseq
ResNet50: startseq একটি রোদ্রোজ্জ্বল দিনে একটি ছেলে endseq
InceptionV3: startseq একটি ছেলে ক্ষুদ্র সাঁতারের পোশাক পরে একটি পুলে ঝাঁপ দিচ্ছে endseq
EfficientNet: startseq এক বাচ্চা সুইমিংপুলে ঝাপ দিচ্ছে endseq
Ensemble VGG16: startseq একটি শিশু পুলে ঝাঁপিয়ে পড়ছে endseq
Ensemble EfficientNet: startseq এক বাচ্চা সুইমিংপুলে ঝাপ দিচ্ছে endseq

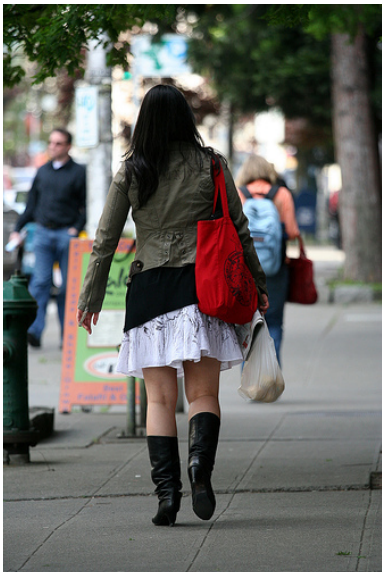


Figure 4.8: Flickr8k: Test image 2

Actual captions:
startseq সাদা স্কার্ট এবং কালো বুটজুতা পরিহিত একজন কালো চুলের নারী একটি লাল থলে হাতে নিয়ে রাস্তা দিয়ে হেঁটে যাচ্ছে endseq
startseq সাদা শার্ট,কালো জুতা এবং ধূসর সবুজ জ্যাকেট পরিহিত একটি খাটো মেয়ে একটি লাল ব্যাগ নিয়ে ফুটপাতে হাঁটছে endseq
startseq ফুটপাথ দিয়ে এক মহিলা হাটছেন endseq
startseq লাল স্কার্ট এবং বুট পরা একজন মহিলা লাল ব্যাগ নিয়ে ফুটপাতে হাঁটছেন endseq
startseq ছোট সাদা স্কার্ট, কালো বুট লাল ব্যাগ পড়া এক নারী হাটছে endseq

Predicted captions: VGG16: startseq এক মহিলা ফুটপাতের পাশে হেটে যাচ্ছে endseq
VGG19: startseq রূপালী পোশাক এবং রূপালী ব্যাগযুক্ত নারী রূপালী ব্যাগ হাতে নিয়ে দাঁড়িয়ে আছেন endseq
ResNet50: startseq লাল রঙের পোশাক পরিহিত একজন নারী তার হাত দিয়ে একটি লাল রঙের খেলনা ধরে আছে endseq
InceptionV3: startseq ইয়ারফোন পরা এক মহিলা ট্র্যাফিক শঙ্কু পেরিয়ে হাঁটছেন endseq
EfficientNet: startseq একটি পার্স ব্যাগ বহন করছে এক মেয়ে endseq
Ensemble VGG16: startseq এক মহিলা ফুটপাতের পাশে হেটে যাচ্ছে endseq
Ensemble EfficientNet: startseq একটি পার্স ব্যাগ বহন করছে এক মেয়ে endseq

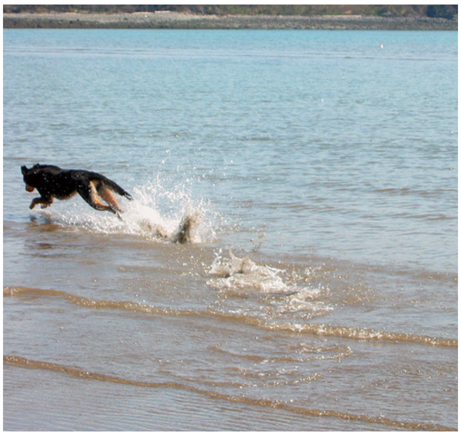


Figure 4.9: Flickr8k: Test image 3

Actual captions:
startseq একটি কালো রঙের কুকুর পানি ছিটিয়ে ছুটে চলেছে endseq
startseq একটি বাদামী এবং ট্যান কুকুর অগভীর জলের মধ্য দিয়ে চলছে endseq
startseq সমুদ্রের পাড়ে একটি কুকুর দৌড়াচ্ছে endseq
startseq একটি কুকুর জলের মধ্য দিয়ে চলছে endseq
startseq কুকুরট পানির মধ্যদিয়ে ছুটছে endseq

Predicted captions: VGG16: startseq একটি কালো কুকুর পানিতে দৌড়াচ্ছে endseq
VGG19: startseq একটি কালো কুকুর সমুদ্রে দৌড়াচ্ছে endseq
ResNet50: startseq একটি কুকুর পানিতে ঝাপ দিচ্ছে endseq
InceptionV3: startseq একটি কুকুর পানিতে তরঙ্গের দিকে তরঙ্গের দিকে এগিয়ে যাচ্ছে endseq
EfficientNet: startseq একটি কুকুর পানিতে দৌড়াচ্ছে endseq
Ensemble VGG16: startseq একটি কালো কুকুর পানিতে দৌড়াচ্ছে endseq

Ensemble EfficientNet: startseq একটি কুকুর পানিতে দৌড়াচ্ছে endseq

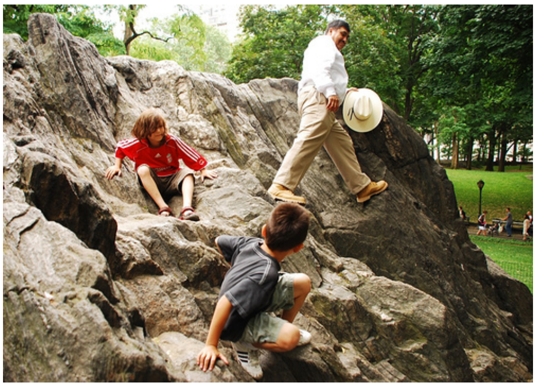


Figure 4.10: Flickr8k: Test image 4

Actual captions:
startseq একদল মানুষ পাথরে হাটছে endseq
startseq একটি লোক এবং দুটি বালক পার্কে একটি পাথর আরোহণ করছে endseq
startseq তিনটি লোক একটি বড় পাথরে আরোহণ করে endseq
startseq দুটি বালক এবং একটি ব্যক্তি একটি সাদা টুপি ধারণ করে একটি পাথরের উপরে উঠেছে একটি লন এবং ছায়া গাছের পটভূমিতে দৃশ্যমান endseq
startseq দু'জন শিশু এবং একটি লোক বোল্ডারে endseq

Predicted captions: VGG16: startseq একজন লোক একটি পাথর আরোহন করছে endseq
VGG19: startseq এক লোক পাথুরে পাহাড়ে আরোহণ করছে endseq
ResNet50: startseq দু'জন লোক একটি পাথরে আরোহণ করছে endseq
InceptionV3: startseq দু'জন লোক একটি পাহাড়ে বসে আছেন endseq
EfficientNet: startseq এক লোক পাথর চড়ছেন endseq
Ensemble VGG16: startseq একজন লোক একটি পাথর আরোহন করছে endseq
Ensemble EfficientNet: startseq এক লোক পাথর চড়ছেন endseq

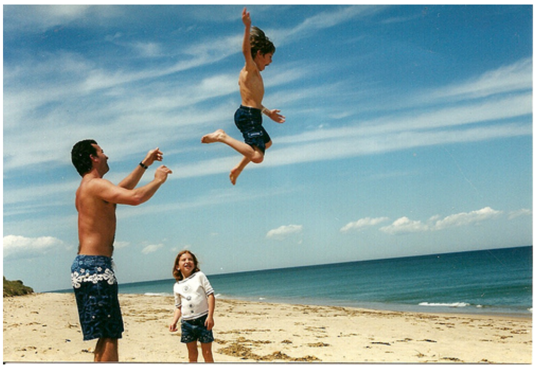


Figure 4.11: Flickr8k: Test image 5

Actual captions:
startseq একজন বাবা তার দুই শিশুকে সাথে নিয়ে সমুদ্রের ধারে খেলা করছে এবং তিনি তার বাচ্চা ছেলেকে শূন্যে ছুঁড়ে দিয়েছেন endseq
startseq এক লোক এক বাচ্চাকে সৈকতে লাফ দেওয়াচ্ছেন endseq
startseq একজন লোক সৈকতে বাতাসে একটি ছেলেকে ছুঁড়ে মারছে endseq

startseq একজন প্রাপ্তবয়স্ক ব্যক্তি একটি সৈকতে বাচ্চাটিকে বাতাসে ফেলে দিচ্ছেন, অন্য শিশুটি দেখছে endseq
startseq কিছু মানুষ সৈকতে খেলছে endseq

Predicted captions:
VGG16: startseq একটি ছেলে সৈকতে লাফ দিচ্ছে endseq
VGG19: startseq এক লোক সৈকতে লাফাচ্ছে ensdeq
ResNet50: startseq এক বাচ্চা পানিতে লাফাচ্ছে endseq InceptionV3: startseq একটি ছেলে সৈকতে রংধনু হ্যান্ডস্ট্যান্ড করছে endseq
EfficientNet: startseq এক লোক সৈকতে লাফাচ্ছে endseq
Ensemble VGG16: startseq একটি ছেলে সৈকতে লাফ দিচ্ছে endseq
Ensemble EfficientNet: startseq এক লোক সৈকতে লাফাচ্ছে endseq

4.2.2 Flickr30k

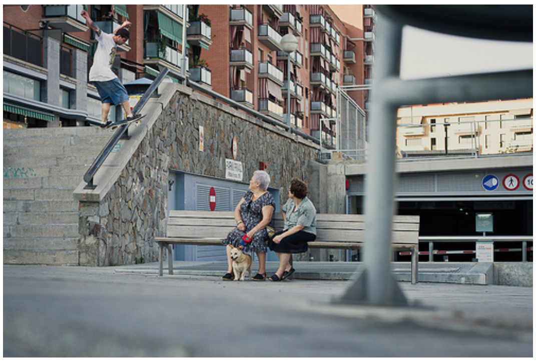


Figure 4.12: Flickr30k: Test image 1

Actual captions:
startseq একটি স্কেটবোর্ডার, নীল শর্টস এবং একটি সাদা টি-শার্ট, একটি রেলিং থেকে নিচের দিকে পিছলে যাচ্ছে যখন দুটি বয়স্ক মহিলা দেখছে, একজন একটি কুকুর ধরে আছে, পার্কের বেঞ্চে বসে আছে। endseq
startseq একটি সাবওয়ের সামনের একটি শহরে, একজন লোক একটি সিঁড়ি দিয়ে স্কেটবোর্ডে নেমে যাচ্ছে যখন একটি বেঞ্চে বসে থাকা দুই বয়স্ক মহিলা তাকে দেখছেন। endseq
startseq দুই বয়স্ক মহিলা, একজন কুকুরের সাথে বেঞ্চে বসে, রেলিংয়ের নিচে একজন পুরুষ স্কেটবোর্ড দেখছেন endseq
startseq দুই মহিলা একটি ছেলে স্কেটবোর্ডের দিকে তাকিয়ে আছে endseq
startseq একটি বাচ্চা একটি স্কেটবোর্ডে চড়ছে endseq

Predicted captions: VGG16: startseq একজন লোক একটি দোকানের সামনে একটি বেঞ্চে বসে আছে endseq

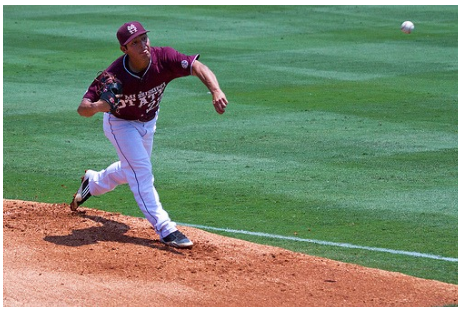


Figure 4.13: Flickr30k: Test image 2

Actual captions:
startseq বেগুনি এবং সাদা ইউনিফর্ম পরা একটি বেসবল কলসি ঢিবি থেকে একটি বেসবল নিক্ষেপ করছে endseq
startseq কলসিটি দক্ষতার সাথে ঢিবি থেকে ব্যাটারের দিকে বল ছুঁড়ে দিল endseq
startseq মেরুন ক্যাপ এবং জার্সিতে পিচিং মাউন্ডের উপর কলস বেসবল প্রকাশ করে endseq
startseq লাল এবং সাদা ইউনিফর্ম পরা এক যুবক বেসবল নিক্ষেপ করছে endseq
startseq একজন বেসবল খেলোয়াড় একটি বেসবল নিক্ষেপ করছে endseq

Predicted captions: VGG16: startseq একজন বেসবল খেলোয়াড় একটি পিচ নিক্ষেপ করছে endseq

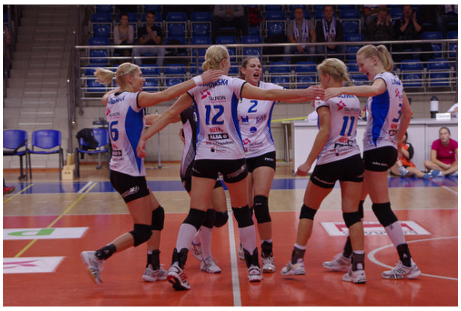


Figure 4.14: Flickr30k: Test image 3

Actual captions:
startseq হাঁটুর প্যাডে ছয়জন অ্যাথলেটিক মেয়ে একে অপরের দিকে এগিয়ে যাচ্ছে যখন দর্শকরা তাদের আসন থেকে দেখছে endseq
startseq একটি মেয়েদের ভলিবল দল একটি বিরল দর্শকদের সামনে উত্তেজিতভাবে আড্ডা দিচ্ছে। endseq
startseq বেশ কিছু মেয়ে ইউনিফর্ম পরে ভলিবল ম্যাচে অংশ নেয় endseq
startseq একদল মহিলা ক্রীড়াবিদ একসঙ্গে জড়াজড়ি করে এবং উত্তেজিত endseq
startseq আড্ডায় একটি মেয়েদের ভলিবল দল endseq

Predicted captions: VGG16: startseq একটি মেয়েদের ভলিবল দল একটি খেলা খেলছে endseq

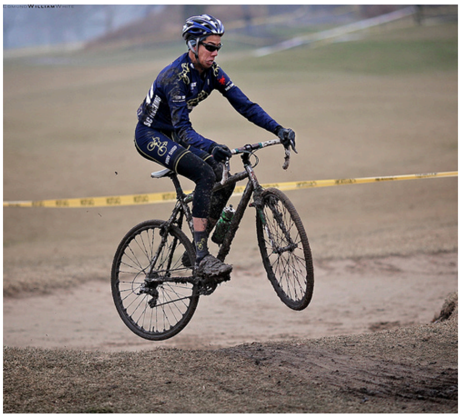


Figure 4.15: Flickr30k: Test image 4

Actual captions:
startseq একটি সাইক্লিং স্যুট এবং বাইকের হেলমেট পরা একজন লোক একটি নোংরা ট্র্যাকে বাইক চালাচ্ছেন। endseq
startseq মাথা থেকে পা পর্যন্ত কাদায় ঢেকে থাকা বাইকে এক ব্যক্তি endseq
startseq একজন কর্দমাক্ত ব্যক্তি একটি কোর্সের মধ্য দিয়ে তার সাইকেল চালাচ্ছেন endseq
startseq একজন বাইকার মাঝ আকাশে স্টান্ট করছে endseq
startseq একজন লোক তার বাইকে চালাকি করছে endseq

Predicted captions: VGG16: startseq একজন লোক একটি ময়লা বাইকে একটি কৌশল করছে endseq

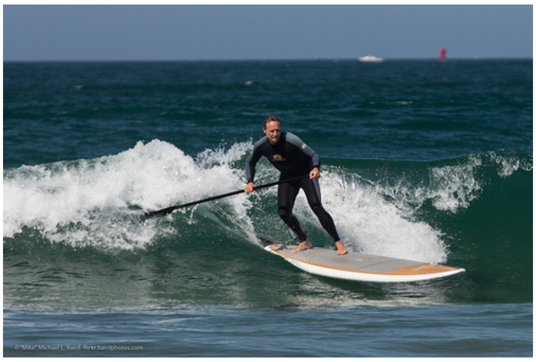


Figure 4.16: Flickr30k: Test image 5

Actual captions:startseq একজন লোক একটি মাঝারি আকারের তরঙ্গে সার্ফ করছে যখন একটি প্যাডেল ধরে আছে যা তাকে সে যে দিকে যেতে চায় সেদিকে চালনা করছে endseq
startseq ওয়েটস্যুট পরা একজন প্যাডেল বোর্ডার তীরে ঢেউ চালাচ্ছে endseq
startseq ওয়েটস্যুট পরা একজন ব্যক্তি সার্ফ প্যাডেল ব্যবহার করে সার্ফিং করছেন endseq
startseq একজন সার্ফার একটি বড় সার্ফবোর্ডে ঢেউ চালাচ্ছেন endseq
startseq কালো পরা লোকটি একটি ঢেউ সার্ফ করছে endseq

Predicted captions:
VGG16: startseq একজন সার্ফার একটি তরঙ্গে চড়ছে endseq

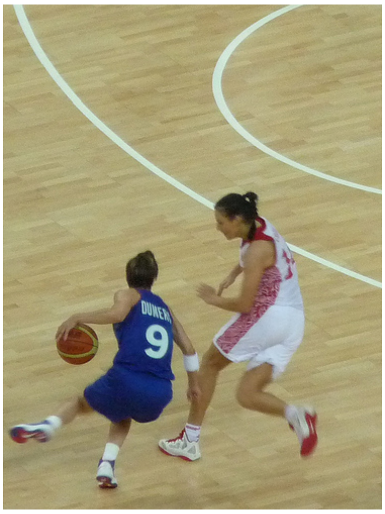


Figure 4.17: Flickr30k: Test image 6

Actual captions:
startseq বাস্কেটবল খেলার সময় সাদা ইউনিফর্ম পরা নারীদের কাছ থেকে বলটি রক্ষণাত্মকভাবে দূরে রাখেন নম্বর জার্সি পরা নারী endseq
startseq দুই মহিলা, বিরোধী বাস্কেটবল দলে, কোর্টে মুখোমুখি endseq
startseq বাস্কেটবল ইউনিফর্ম পরা দুই মহিলা কোর্টে বাস্কেটবল খেলছেন endseq
startseq বিভিন্ন দলের দুই মহিলা বাস্কেটবল খেলছে endseq
startseq দুই মহিলা বাস্কেটবল খেলা খেলছেন endseq

Predicted captions:
VGG16: startseq স্পেশাল অলিম্পিকে ভলিবলে স্বর্ণপদকের জন্য প্রতিদ্বন্দ্বিতা করছে দুই নারী endseq

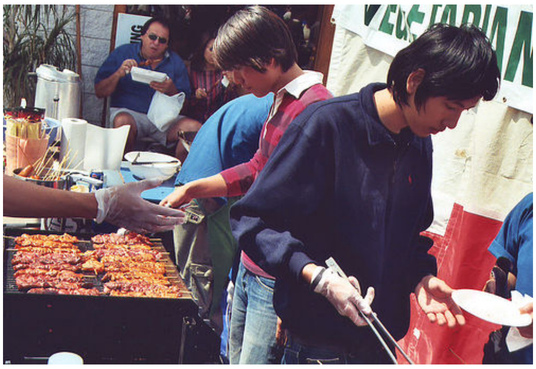


Figure 4.18: Flickr30k: Test image 7

Actual captions:
startseq ক্ষীরের গ্লাভস পরা একটি নীল শার্ট পরা এবং চিমটা ধরে থাকা একজন ব্যক্তি সহ বেশ কয়েকজন লোক একটি রান্নার জায়গার চারপাশে দাঁড়িয়ে আছে যেখানে মাংস রান্না করা হচ্ছে endseq
startseq দুই যুবক একটি গ্রিলের উপর কাজ করছে, লাঠিতে খাবার তৈরি করছে এবং পরিবেশন করছে endseq
startseq রাস্তার বিক্রেতারা গ্রিলড খাবার তৈরি এবং পরিবেশন করছে endseq
startseq পুরুষরা গ্রিল থেকে রান্না করে খাবার পরিবেশন করছে endseq
startseq পুরুষরা গ্রিল দিয়ে খাবার পাচ্ছে endseq

Predicted captions:
VGG16: startseq একজন লোক একটি গ্রিলের উপর মাংস গ্রিল করছে endseq

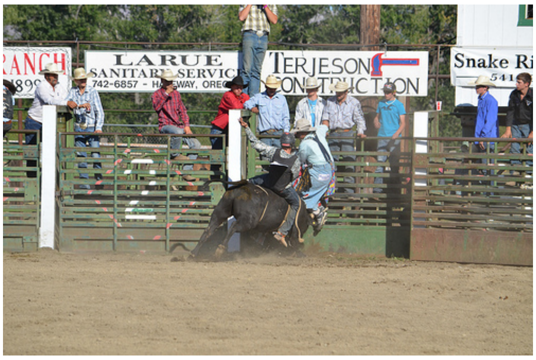


Figure 4.19: Flickr30k: Test image 8

Actual captions:
startseq ধাতুর বেড়ার উপর রাখা কাউবয়দের দ্বারা বেষ্টিত, একজন যুবক ষাঁড়ের চড়ার প্রতিযোগিতায় অ্যাঙ্গাস ষাঁড় থেকে প্রায় পড়ে যাচ্ছে endseq
startseq একটি রোডিও ষাঁড় রাইডার এবং একজন সাহায্যকারীর বাইরের রোডিও শোতে যতটা সামলাতে পারে তার চেয়ে বেশি কিছু আছে। endseq
startseq একজন আরোহী একটি রোডিওতে ষাঁড়ে চড়ে ভিড় দেখছে endseq
startseq জন লোক একটি ষাঁড়ে চড়ে বেড়াচ্ছেন যখন মানুষের ভিড় তা দেখছে endseq
startseq একজন লোক একটি রোডিওতে ঘোড়ায় চড়ে endseq

Predicted captions:
VGG16: startseq একজন লোক একটি রোডিওতে একটি ষাঁড়ে চড়ে endseq

4.3 Evaluating Results

We have validated the accuracy of the generated captions. We are using the BLEU (BiLingual Evaluation Understudy) scores of the model for different CNN architectures as a metric for accuracy. BLEU scores are used to evaluate the quality of translated or generated text from a natural language by a machine. Generally, a BLEU score of 0.4-0.5 indicates a decent result [16].
The BLEU scores attained are shown below:

	VGG16	VGG19	ResNet50	Inception V3	Ensemble VGG16	Effecient NetB3	Ensemble Effecient NetB3
BLEU-1	0.512774	0.513210	0.537591	0.511825	0.512711	0.553197	0.550006
BLEU-2	0.308157	0.307073	0.331672	0.307585	0.308161	0.341976	0.337606
BLEU-3	0.209269	0.210217	0.233927	0.206523	0.209280	0.234436	0.237218
BLEU-4	0.098938	0.095334	0.112947	0.090928	0.098947	0.113089	0.119143

Table 4.1: BLEU scores for various CNN architectures

4.4 Discussion

We have trained all the models for 25 epochs. It seemed like the ideal number of iterations because upon training the model for fewer and more epochs, we have noticed either no notable improvement or worsening accuracy. A dropout layer of value 0.4 has been chosen as lower or higher value was causing important features to be filtered out or letting in noise and unnecessary data. We have tried using a the sgd and rmsprop optimizers for our model but they were not as efficient as Adam.

The best CNN model for our purpose is the EfficientNetB3 model. It has the highest BLEU scores among all others and the captions generated are very satisfactory.

Even though some variations of the model have achieved a higher BLEU score than others, it does not necessarily mean that it can provide better captions. For example, the model using ResNet50 has a higher BLEU score than the ones using VGG16. ResNet50 has been tested to have a higher accuracy than VGG16 on a specific dataset. But for our Flickr8k dataset, VGG16 performed better. Therefore, we be certain that how good a CNN model performs depends greatly on the dataset being used.

If a datasets lacks all possible kinds of data, it will affect the training of the model. The model cannot describe the image properly in that case. In the Flickr8k dataset, there are more images containing dogs than other animals. When our model was given an image of cats to annotate, it mistakenly said that dogs are present in the image. There are also images that are misleading to CNN architectures. The model may mistakenly classify silhouettes in an image as objects or animals but not people. Too many images portraying the same objects, environment or actions can make the model biased towards them. Such circumstances can affect the accuracy and performance of a model.

Another thing we have realized is that models trained with translated captions for the images in Flickr30k have performed as well morels trained with original Bangla captions. The generated captions have been meaningful and coherent. However, we were only able to train models with captions translated by Google translate. We were unable to use captions translated by BanglaNLG because of hardware limitations. We were facing a resource exhaustion error that we were unable to solve. Aside from this, the captions translated by this service were not very accurate. Many of the captions were meaningless in the context of the image. This makes sense because it is a rather recent model developed by the students of Bangladesh University of Engineering and Technology and it requires more research to improve.

Chapter 5

Challenges

We used Flickr8k and Flickr30k datasets as image data. Both the Flickr8k and Flickr30k datasets are frequently utilized in the field of computer vision and natural language processing for activities such as image captioning. Both of these datasets were collected by Flickr. Both sets of data are made up of photographs that have captions written next to them.

Both the Flickr8k and Flickr30k datasets have a total of 8,091 and 31,794 photos, respectively. Both datasets have been utilized to a large extent in a variety of research projects and have been demonstrated to be useful resources for the training and evaluation of image captioning models.

The Flickr databases provides five different captions for each image so that a variety of different descriptions can be provided for each picture. Human annotators were tasked with describing the contents of the image using a natural language framework while creating these captions. The Flickr database makes it possible to create image captioning models that are more robust and accurate because to the fact that it provides many captions for each image. For instance, a model trained on the Flickr database, which contains five captions for each image, may be able to generate captions for a specific image that are more diverse and descriptive than those that can be generated by a model trained on a dataset that contains only one caption for each image. Since the model has access to various descriptions of the same image, using several captions for a single image can also make it possible to create models that are more resistant to overfitting. This is because the model has more information to work with.

We took Bangla datasets, each of which contained five different descriptions for each image, much like we did with the English database. When dealing with Flickr8k, we made use of the BAN-CAP caption database, which contains captions that have already been annotated. Unfortunately, we were unable to find any Bangla annotated caption databases for the Flickr30k platform. As a result, we utilized two different strategies. The first option was to translate the captions using Google Translate which game us decent translations for each captions. The other is using English to Bangla translation model, which was provided by the Bangladesh University of Engineering and Technology. We built a code to translate our captions after forking their model, which was located on their GitHub page under the name "csebuetnlp." The translation was carried out by a machine, which meant that it did not take into account any of the details of the scenario. This led to conclusions that were not entirely reliable in either instance. In addition, the "csebuetnlp" translation model frequently produced translations that were both unrealistic and invalid. This not only caused the length of the sentences to double, but it also prevented the computers from learning from the captions due to longer sentences than their limit. Dealing with the variable nature of the images can be one of the challenges encountered when attempting to extract characteristics from photos contained within these datasets. Variations in lighting and background, as well as the presence of many items in the same shot, are examples of this. Because of these variances, it might be challenging for models to correctly recognize and describe the items that are depicted in the photographs.

When working with huge datasets such as Flickr30k, you run the risk of encountering graph execution failures while attempting to train the dataset on a local workstation. This is just one of the many challenges that come with working with such datasets. It's possible that the machine doesn't have enough memory or processing capacity to prevent these kinds of problems from happening. It may be essential to utilize a machine that has more powerful hardware or to use a cloud-based service that has access to more resources in order to resolve these issues. Both of these options are available.

In conclusion, the Flickr8k and Flickr30k datasets are extensively utilized for picture captioning jobs; nevertheless, it can be difficult to extract characteristics from the photos due to the high level of variability present in the images. In addition, training on a local system with big datasets such as Flickr30k might lead to graph execution failures, which may necessitate the use of more powerful hardware or a solution that is hosted in the cloud.

Chapter 6

Conclusion

6.1 Future Works

We hope that in the future, individuals from all walks of life will be able to benefit from our Bangla image caption generator. We hope that the media, social services, schools, businesses, law enforcement, and other sectors will find our Bangla image caption generator useful.

Initially, to aid the visually challenged, it would be helpful to have an online image caption generator platform that can be installed on cross-platform devices such as phones, laptops, or even embedded devices. For example, it will be used to caption photographs to help the visually impaired understand the content of photos shot with a mobile device's camera. This technology might be used to automatically generate captions about what is being shown on the screen too. For the visually impaired, this system might also be used to provide descriptions for web photos. People with visual impairments would benefit greatly from this platform, which might give them the freedom of understanding which was previously impossible.

Furthermore, our Bangla image caption generator will be useful in the future for assisting people in various sectors in gaining a deeper appreciation for the content of visual media. In the medical field, for instance, it can be used to help doctors and patients in the rapid recognition and diagnosis of medical disorders depicted in patient photos. It is also useful in the classroom since they give students context for the pictures they're looking at. It also has many practical applications in social life, like the detection of faces and the fast identification of common objects which again will help the visually impaired as well as foreigners who are unaware of the scene or objects. Improving the accuracy and efficiency of image captioning algorithms and technologies is essential for all of these uses.

In addition to the medical, educational, and social fields, image caption generators can also be used in other areas such as the legal field. For example, image caption generators can be used to help lawyers quickly identify evidence in images or videos that could be used in court cases. In the business field, image caption generators can be used to help companies quickly identify products or services in images for marketing purposes. In the entertainment industry, image caption generators can be used to help viewers better understand and interpret visual media such as movies or television shows. All of these applications will require further development of image captioning algorithms and technologies to make them more accurate and efficient.

6.2 Conclusion

There are some Bangla caption generation options available now that can help us overcome the challenges of creating native objects and their exact captions. However, no individual method has proved completely accurate, given the image caption generation concept is less than a decade old. We are concentrating on correcting the probability and developing new ways to fine-tune the production of Bengali image captions. As a result, the report has been written to solve specific problems and generate appropriate solutions for any case. As a result of this, the outputs that are created are meaningful and nearly accurate in the majority of instances. However, it still has a long way to go while the technology advances and it will reach its full potential.

Bibliography

[1] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, 2014. DOI: 10.48550/ARXIV.1409.1556. [Online]. Available: https://arxiv.org/abs/1409.1556.

[2] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, 2015. DOI: 10.48550/ARXIV.1512.03385. [Online]. Available: https://arxiv.org/abs/1512.03385.

[3] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, *Rethinking the inception architecture for computer vision*, 2015. DOI: 10.48550/ARXIV.1512.00567. [Online]. Available: https://arxiv.org/abs/1512.00567.

[4] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015.

[5] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016.

[6] M. Aiken, "An updated evaluation of google translate accuracy," *Studies in Linguistics and Literature*, vol. 3, p253, Jul. 2019. DOI: 10.22158/sll.v3n3p253.

[7] M. Rahman, N. Mohammed, N. Mansoor, and S. Momen, *Procedia Computer Science*, vol. 154, pp. 636–642, 2019, Proceedings of the 9th International Conference of Information and Communication Technology [ICICT-2019] Nanning, Guangxi, China January 11-13, 2019, ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2019.06.100. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050919308701.

[8] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," 2019. DOI: 10.48550/ARXIV.1905.11946. [Online]. Available: https://arxiv.org/abs/1905.11946.

[9] A. Jain, H. Patel, L. Nagalapatti, *et al.*, "Overview and importance of data quality for machine learning tasks," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery amp; Data Mining*, ser. KDD '20, Virtual Event, CA, USA: Association for Computing Machinery, 2020, pp. 3561–3562, ISBN: 9781450379984. DOI: 10.1145/3394486.3406477. [Online]. Available: https://doi.org/10.1145/3394486.3406477.

[10] A. H. Kamal, M. A. Jishan, and N. Mansoor, "Textmage: The automated bangla caption generator based on deep learning," in *2020 International Conference on Decision Aid Sciences and Application (DASA)*, 2020, pp. 822–826. DOI: 10.1109/DASA51403.2020.9317108.

[11] M. Faiyaz Khan, S. M. Sadiq-Ur-Rahman, and M. Saiful Islam, “Improved bengali image captioning via deep convolutional neural network based encoder-decoder model,” in *Proceedings of International Joint Conference on Advances in Computational Intelligence*, M. S. Uddin and J. C. Bansal, Eds., Singapore: Springer Singapore, 2021, pp. 217–229, ISBN: 978-981-16-0586-4.

[12] M. A. H. Palash, M. A. A. Nasim, S. Saha, F. Afrin, R. Mallik, and S. Samiappan, “Bangla image caption generation through cnn-transformer based encoder-decoder network,” in *arXiv*, 2021. DOI: 10.48550/ARXIV.2110.12442. [Online]. Available: https://arxiv.org/abs/2110.12442.

[13] A. Bhattacharjee, T. Hasan, W. U. Ahmad, and R. Shahriyar, *Banglanlg: Benchmarks and resources for evaluating low-resource natural language generation in bangla*, 2022. DOI: 10.48550/ARXIV.2205.11081. [Online]. Available: https://arxiv.org/abs/2205.11081.

[14] *Ban-cap: English-bangla image descriptions dataset | kaggle*, https://www.kaggle.com/datasets/saifsust/bancap, (Accessed on 09/15/2022).

[15] *Bangla language - banglapedia*, https://en.banglapedia.org/index.php/Bangla_Language, (Accessed on 09/18/2022).

[16] *Evaluating models*, https://cloud.google.com/translate/automl/docs/evaluate, (Accessed on 01/12/2023).

[17] *Flickr30k: English-bangla image descriptions dataset | kaggle*, https://www.kaggle.com/datasets/hsankesara/flickr-image-dataset, (Accessed on 05/01/2022).

[18] *Flickr8k: English-bangla image descriptions dataset | kaggle*, https://www.kaggle.com/datasets/adityajn105/flickr8k, (Accessed on 05/01/2022).