

# Eyes Through Words: Providing Independence to Visually Impaired with Photo-to-Text Technology

M.Sc. Research Project  
M.Sc. Data Analytics

Sonia.

Student ID: 21150192

School of Computing  
National College of Ireland

Supervisor: Vladimir Milosavljevic

National College of  
Ireland Project  
Submission Sheet  
School of  
Computing



<b>Student Name:</b>	Sonia
<b>Student ID:</b>	21150192
<b>Programme:</b>	Research Project
<b>Year:</b>	2022-2023
<b>Module:</b>	MSc Research Project
<b>Supervisor:</b>	Vladimir Milosavljevic
<b>Submission Due Date:</b>	14/08/2023
<b>Project Title:</b>	Eyes Through Words: Providing Independence to Visually Impaired with Photo-to-Text Technology
<b>Word Count:</b>	10966
<b>Page Count:</b>	30

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

**ALL** internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

<b>Signature:</b>	Sonia
<b>Date:</b>	14th August 2023

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:**

Attach a completed copy of this sheet to each project (including multiple copies).	<input checked="" type="checkbox"/>
<b>Attach a Moodle submission receipt of the online project submission</b> , to each project (including multiple copies).	<input checked="" type="checkbox"/>
<b>You must ensure that you retain a HARD COPY of the project</b> , both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input checked="" type="checkbox"/>

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

<b>Office Use Only</b>	
Signature:	
Date:	
Penalty Applied (if applicable):	

## Abstract

Captioning images automatically is a difficult problem that spans both computer vision and natural language processing. Recent attention has been spurred by the importance of this job in applications like as self-driving cars, helping the visually handicapped, and improving the detection of malicious activities. Captioning photos of cricket matches is of particular interest in this research. Approximately a thousand pictures of cricketers were gathered and labelled with an average of two captions each. We have personally collected cricketer images and generated associated labels, making this dataset a valuable contribution to our research. The created model for cricketer picture captioning was compared against the popular Flickr dataset, which consists of 8,000 photographs with five captions each. The study used a Long Short-Term Memory (LSTM)-based Recurrent Neural Network (RNN) to process textual captions, and a pre-trained InceptionV3 model to extract picture data without the classification layer. An image feature extractor, sequence processor, and decoder make up the model's architecture. The accuracy of the predicted captions was compared to the accuracy of the reference captions using BLEU ratings. Both BLEU-1 (uni-gram scoring) and BLEU-2 (bi-gram scoring) were used in the model selection process. Caption loading, caption prediction for new photos, and other implementation issues are discussed in length. While the cricket captioning model performed admirably on cricketer-related photos, it struggled when applied to unrelated images. However, on photographs of cricketers, it beat a model trained using data from Flickr. In contrast, the Flickr-trained model performed exceptionally well over a wide range of domains since it was built on a more comprehensive dataset.

**Keywords:** Automatic Caption Generation, computer vision, natural language processing, visual assistants, sports-related captioning, cricket images, recurrent neural network, bias-variance estimation units (BLEUs), model selection, attention-based modules, hyper-parameter tuning for large CNN models, and image feature vectors.

## Chapter 1. Introduction

Automatic image captioning means providing textual description of an image or the content observed in that image. To explain in little more details, observe below image Fig1:



**Figure 1.** Cricket field image (ref: Google-search)

It's possible that after looking at Figure 1, one of us will remark, "A fielder is attempting to catch a cricket ball." Different people will describe the same action in different ways: "A fielder is diving to have a catch on grassy field," and "A player dressed in green and yellow is diving to have a catch." We are well aware that the previous captions are all extremely relevant to Fig.1. In addition, we are cognizant of the fact that this is a really easy task for human beings. In order to provide a detailed description of an image, we just need to peek at it briefly. A little youngster of around six years of age would have no trouble at all with this task. However, a machine or computer would have a hard difficulty making sense of that image.

## 1.1 Motivation

Even the most accomplished computer vision specialists had difficulty solving this challenge until the advent of deep neural networks in recent years. But with deep learning's progress, we can easily do this task if we can locate the necessary datasets. This problem was very much researched by Karpathy, A., & Fei-Fei, L., 2015. They researched this topic in great details. Karpathy, A. [2015], explained how deep learning models, image data and respective textual descriptions can be used to develop an end-to-end image captioning model. Since then, image captioning has been an interesting topic and received a very good amount of attention in past few years because of its importance in real life applications. Particularly, an image captioning model has two types of methods where it will require computer vision models to understand the content like corners, edges, density etc. in an image and a language model or natural language processing model will be required to understand the textual description or caption in proper order. In recent times, deep learning models have state-of-art results on examples of image captioning problem. The most interesting thing about these models is single end-to-end deep model architectures can be defined to predict image caption when a photo is loaded without need of sophisticated data pre-processing or a pipeline of some specific models.

## 1.2 Aim

The purpose of this study is to explain that how computer vision and natural language processing can be used to solve image captioning problem given an image as input. First, we should understand how important this problem is some real life applications. Let's explore some of the use-cases where automatic image captioning can be significantly useful. **Visual assistance** to blind/needy people: It can be implemented in a product that will guide them on public places. It can be done by first creating the scene into textual description and then text data to audio speech. Now, both tasks are quite famous in deep learning. Currently NVIDIA research team is working on such product. **Crime reduction**: Nowadays, CCTV camera are around most of the places. If we can implement image captioning based alarms that will help to reduce crime or malicious activities. **Tuning google image search**: Automatic image captioning can improve google image search as excellent as google search. Once we upload the images, automatic captions will be generated and captions related images or content will be searched. **Self-driving cars**: Automatic vehicle driving is one of the biggest challenges. If automatic captioning can elaborate the scene near to car, it will be a great boost for self-driving car projects. Some of other use-cases are caption suggestion during uploading the image, indexing an image etc.

## 1.3 Research Hypothesis

This study is conducted to fulfil following objectives:

**RQ1**: How to develop an image captioning model on any sport specific data that can help in giving a briefing happening in videos/images?

**RQ2:** How to build image captioning model on some standard data to compare the performance of sport-specific captioning?

**RQ3:** How to make a system that will help the blind people and for other use cases? **RQ4:** What are the important algorithm performance analysis found out?

Further chapters of report are structured as follows: Next section of this chapter presents the objective of our study. Chapter 2 describes the literatures or research papers considered/referred for this study. Chapter 3 describes the methodology used in current study including details of standard CNN architecture InceptionV3 model, LSTM model, GloVe model, BLEU scores etc. Chapter 4 presents the implementation of our study, which explains dataset preparation, loading image & captions, pre-processing images and captions, preparing train samples, model fitting, model selection, model testing & prediction on new unseen images. Chapter 5 puts forward the results of both cricket captioning model & Flickr-trained model and provide the comparisons, model evaluation results & other insights related to model & data etc. Chapter 6 presents the summary and findings of present study and scope of future work. References for the project work are presented in the last section.

## **Chapter 2: Literature Review**

To attain the objectives of our research on the concepts of Multimodal learning in the most efficient way we can, we would do a literature review of 21 research papers published by many reputed authors who have a lot of expertise in the field of Machine Learning, especially Multimodal learning. Some of the research articles which we would use as references for our study are as follows:-

### **2.1 Research done in this field in the aspect of Multimodal learning**

The results of a unique shared job in Natural Language Processing and Computer Vision are presented in (Specia, L., Frank, S., Sima'an, K., & Elliott, D., 2016). The work involves generating picture descriptions in a target language from an image and/or descriptions in a source language. This task requires taking an input image and utilising it to generate textual descriptions in the target language. The tournament, which took place during WMT16, had tasks including creating translations and summaries. A total of sixteen translation systems and seven description systems were submitted by 10 separate teams.

Martinec and Salway's (2005) study provides a standardised framework of image-text connections for multimodal discourse. This approach integrates the semantic and logical relationships between images and words. For both human and machine analysts, it identifies units, describes each image-text pairing, and explains the logical-semantic and status connections. The algorithm can tell the difference between new and traditional media when it comes to image-text links.

The multitask learning framework was proposed in 2017 by Elliott, D., and Kádár, A. By decomposing multimodal translation into the translation and visual grounding tasks, this paradigm improves translation accuracy. The Multi30K dataset is a good fit for this approach, but other MS COCO datasets might also benefit from it. Training on parallel text and descriptive image datasets has little impact on performance, and improvements to text-only baselines may be made with the help of image prediction. Next steps include expanding the decomposition's scope to encompass other natural language processing issues, trying out other picture prediction architectures, learning new methods for inputting anticipated images, and implementing multitasking at both the encoder and the decoder levels.

In (Mao, J., Xu, W., Yang, Y., Wang, J., & Yuille, A.L., 2014), the authors present a multimodal Recurrent Neural Network (m-RNN) model for generating phrase descriptions for images. Phrase synthesis is handled by a deep recurrent neural network, while image description is handled by a deep convolutional neural network. Both word generation and picture description probabilities are modelled using these two separate networks. The m-RNN model gets a significant performance boost over the state-of-the-art and state-of-the-art generating methodologies when compared on three separate datasets.

According to a report (Langlotz, C.P., Manning, C.D., Miura, Y., & Zhang, Y., 2022) ConVIRT is an unsupervised learning strategy that uses linked descriptive text to learn visual representations in the medical field. This strategy is not domain-specific and does not require the input of experts. It consumes just 10% as much labelled training data as an ImageNet-initiated counterpart does, demonstrating more efficient use of data. It also outperforms strong baselines in most cases. Compared to its ImageNet-initiated counterpart, ConVIRT requires just 10% as much labelled training data.

There is a great opportunity for colleges and universities to create inclusive, engaging classrooms via the use of educational technology. The incorporation of multimedia into classrooms has opened up new avenues for teaching and learning. Students had positive experiences with multimodal learning components, however the findings of an experiment showed that having many representations of the same material did not boost learning performance. This study (Sankey, M., Birch, D., & Gardiner, M.W., 2010) found that teachers should think critically about which representations of essential ideas to include in their lessons. They should pay special attention to the use of audio and video.

Research on the multimodal approach to learning in the setting of the science classroom (Jewitt, C., Kress, G., Ogborn, J., & Tsatsarelis, C., 2001) focuses on the interaction between verbal, visual, and tactile modalities of communication. Students' interests and life circumstances drive the curriculum's emphasis on choosing, adaptation, and change. The goal of this research is to find out if and how analogy was used in the composition of four scientific essays by seventhgraders regarding onion cells.

## **2.2 Research done towards Language Translation using Multimodal methods**

Cognitive behavioural therapy (CBT) employs the techniques of bidirectional translation training and reinforcement learning to assess cross-modal association. Qi, Peng (2018 ed.). Its effectiveness in cross-modal retrieval has been experimentally examined against state-of-the-art techniques on a number of datasets, with encouraging findings.

The second shared goal is multimodal machine translation, which requires the development of 19 separate systems by 9 separate groups. While multimodal systems have made strides, textonly solutions are beginning to offer formidable competition. Both internal and external resources are given equal weight in the evaluation process, as stated by Elliott et al. (2017). This demonstrates the creative possibilities of combining textual and visual elements.

This technique breaks down the steps of multimodal translation into their constituent parts, which are translation and the development of grounded representations. This aids translation because it facilitates the acquisition of many skills at once. The Multi30K and MS COCO datasets attest to its efficacy, and Elliott and Kádár (2017) claim that it is adaptable to improvements in picture prediction and other architectural changes.

With their work on video-text retrieval, Mithun et al. (2018) have made significant strides forward. This approach consists of a fusion mechanism, a modified pairwise ranking loss, and multimodal cues. the superior performance is attributable to the utilisation of multimodal

correlation for enhanced embeddings, as seen in particular in experiments including MSVD and MSR-VTT.

This method primarily focuses on segmented photographs and the relationship between text and those visuals, and it makes use of auto-annotation and the labelling of areas to predict words for whole images as well as individual sections. This is accomplished by examining the degree to which the text and the images are comparable. This approach is useful for both image classification and object detection since it analyses annotated shots. Barnard and coworkers demonstrated this in 2003. Furthermore, it offers additional metrics against which success may be judged.

Multi30K's German translations and crowdsourced descriptions are an expansion of the Flickr30K project. The quirks it contains make it useful for multilingual workloads and multimodal machine translation, but it also creates challenges. The research of Elliott and colleagues suggests that. Its purpose is to encourage community engagement in the creation of different datasets in order to further multilingual and multimodal research.

By analysing the data, Baltruaitis et al. (2018) examine into the impact that deep learning has on multimodal intelligence. Acquisition of representations, fusion of signals, and application development are three possible areas of study. Basic principles like as embeddings and the fusion of different forms of information are explored, along with applications such as the conversion of images to text, the manufacture of images from text, and the answering of visual inquiries. Its goal is to facilitate further investigation into a field that is continually expanding..

### **2.3 Research done in the field of Multimodal learning with Deep Learning**

According to a study by Kiros and coworkers (2014), this work provides a taxonomy of recent developments in multimodal machine learning that goes beyond fusion techniques. As well as tackling wide-ranging issues including representation, translation, alignment, fusion, and colearning, this paradigm is useful for gaining an overall perspective on the field and charting the course for future study.

I. Study by Summaira et al. (2021) Images, movies, texts, sounds, facial expressions, and physiological data are all considered in this analysis of multimodal deep learning. This investigation is meant to include a wide range of modalities, including physical actions, emotional states, and other biometric indicators. In doing so, it sheds light on prior work in the subject as well as future possibilities for application development by providing a thorough categorization of applications, architectures, datasets, and assessment metrics.

Researchers found that using a mix of deep convolutional and recurrent neural networks, the m-RNN model can produce captions for images. In particular, it outperforms state-of-the-art methods in retrieval tasks on standard datasets.

S. Kiros et al. (2014) [Insert citation here] This approach proposes a unified framework by combining the functions of vision and language through the medium of conditional text generation. The ability to learn several tasks concurrently inside a single constraint space, as well as its competitive performance and generalisation, making it an attractive option.

Article by Zhu et al., published in 2017. Using a conditional generative model with a lowdimensional latent vector, the authors of this paper present a solution to the problem of ambiguous image-to-image translation. This method employs a bijective connection to guarantee uniformity between latent encoding modes and output modes, which both reduces the likelihood of mode collapse and yields a more diverse set of realistic results.

Researchers Chen et al., 2020 indicate that UNITER is offered as a general-purpose representation after extensive pre-training on many image-text datasets. Using techniques like optimal transport-based alignment and conditional masking, it achieves state-of-the-art performance across a range of visual-linguistic tasks.

## **2.4 Summary of the papers researched**

The ability to generate image descriptions in target languages and model cross-media imagetext relationships are two indicators of this potential's actualization. Contrastive learning and joint embedding are two examples of the methods that simplify translation work and facilitate the creation of trustworthy medical visual representations. Unified Image-Text Representation (UNITER) and other recent frameworks have shown the promise of using pre-trained models for a wide range of visual-linguistic applications. Accurate and diverse photo captions are being generated thanks, in part, to the study of deep learning, recurrent neural networks, and convolutional networks in multimodal situations. By utilising common benchmarks, evaluations, and surveys, these research help fill in the gaps in our understanding of multimodal machine learning. With this knowledge comes new avenues for research and potential applications.

## **2.5 Research Contributions**

The major goal of this study was to create and test an image captioning model for crickets. On average, two captions were added to each of the roughly one thousand photos of crickets that were collected. We also used the widely-used Flickr8k dataset for comparison, which includes 8000 photos and 40000 captions (5 captions per image). As a standard for picture captioning model creation, the Flickr8k dataset provides uniformly split training, development, and test data. A recurrent neural network (RNN), in particular a long short-term memory (LSTM) model, was used to understand caption sequences, and image feature vectors were extracted to represent the content of the images. A dense layer followed by a Soft-max layer was used to predict upcoming words by combining visual attributes with processed captions. Model selection was based on BLEU scores, with an emphasis on BLEU-1 and BLEU-2 scores, which reflect relative model performance. Models trained with both sets of data were evaluated for accuracy, and the study's findings were written up. The created photo-to-text system has the potential to considerably benefit visually challenged folks, assisting them in comprehending and interacting with the visual environment, therefore the project's consequences go well beyond simply annotating cricket images.

# **Chapter 3. Methodology**

This chapter describes the architecture or methodology used to build an image captioning model using Flickr data or cricket image data and significant sections of this analysis. The major steps used in the methodology are described in next sub-section.

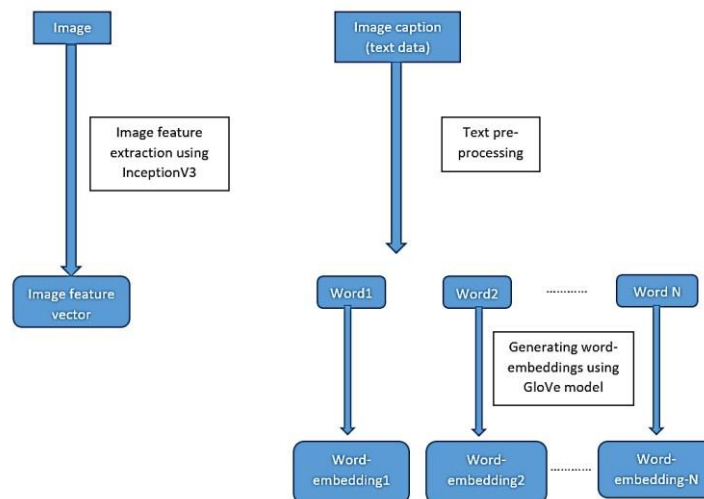
## **3.1 Research Resource – Flowchart of Methodology**

In this chapter we will briefly explain the deep learning models used in our image captioning architecture. Therefore, we won't be going in all the details regarding image captioning implementation here. For cricket data, as we have very small dataset, we will use 800 images as train data and 200 images as test dataset. Although some of the images from cricket data will be filtered out due to non-supported visuals like gif files. In our final cricket captioning model, we will have total 720 images, out of which 575 will be used for model training and remaining 145 will be used for model evaluation (test data). In Flickr dataset, train, development & test



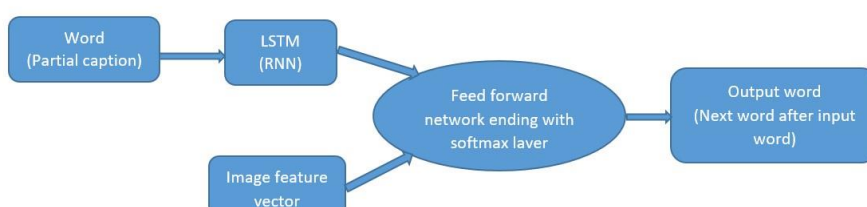
datasets are already segregated as 6000, 1000 & 1000 images and respective captions. Some of below mentioned major steps are structured in Fig 3:

- **The first step** is to import image files which can be in (.jpg, .jpeg etc. formats) and text files with respective captions. We can associate an image with its captions using their common id. Here we know that each image has 5 different captions, so all the captions will be loaded for further steps. Some of the sample image files and respective captions will be presented in one of the next chapters.
- **Second major step** will be used pre-trained InceptionV3 model which is trained on ImageNet dataset. More details regarding InceptionV3 model will be provided when we'll be presented this separately. We will drop last classification layer and remaining model will help us to get 2048-dim vector to represent the visual content of an image.
- **Third major steps** is to get word-embeddings used in image captions. Once data preprocessing like removal of punctuation, numbers, small letter conversion, dropping less frequent words is completed, we will use pre-trained GloVe model to get word-embeddings matrix. GloVe model will be also detailed explained in respective section. This step will give use an embedding-matrix with shape of word-count\*300.



**Figure 2.** Image feature vector and word-embedding extraction

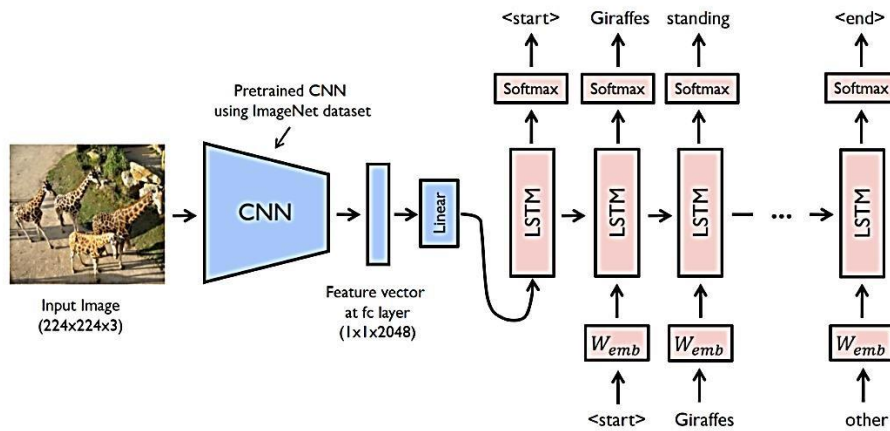
- **Fourth step** is to perform an operation where each word will be passed on recurrent neural network layer supported by long-short-term-memory (LSTM) model. This step is known as sequence processing and will establish the sequential relationship among words. Detailed will be explained in “implementation” chapter with example.
- **Fifth steps** is to merge sequence processor output and image feature vector using a dense layer. This merging layer is known as decoder layer. It will help to relate image and input words with target word. Each time we predict a word that will be added into sequence processor to predict the next word in the sequence using updated word-sequence and image feature vector.



**Figure 3.** Schematic flow of image captioning model

### 3.1.1 Solution Explanation

To get a more feel of image captioning model, we are presenting Fig.5, taken from “AnalyticVidhya” article. It visualizes all the major steps starting from image to getting image feature vector to how LSTM will be implemented to predict the next words etc.



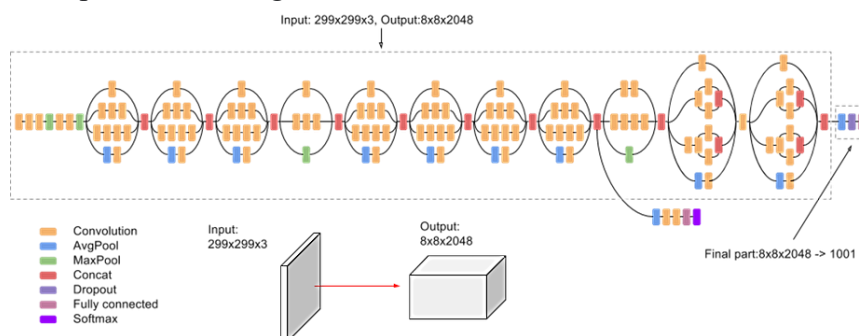
**Figure 4.** Flowchart of image captioning with sample image (ref: Analytics-vidhya)

Once model is fitted after all hyper-parameter tuning and for minimized loss on development data, it will be evaluated on test data. For performance evaluation we will be using BLEU scores known as Bilingual Evaluation Understudy Score. BLEU score generation will be explained in respective section. All above steps will be performed either we are using Flickr & cricket data both or cricket data alone. In the end, major results related insights or findings will be detailed.

## 3.2 Research Method – Machine Learning Models

### 3.2.1 InceptionV3 Model

Convolutional neural networks are very significant state-of-art models to perform any computer vision related tasks. From 2014 onwards, deep convolutional neural networks have been popular, by achieving significant performance on multiple benchmarks. Even though increase in number of parameters of model size and computational complexity improves model performance and task quality, still computational simplicity and small model size are considered very enabling factors to be considered in many use-cases like big-data samples. InceptionNet V3 is built to improve network scaling and utilizing models’ computation efficiency using factorized convolution layers and effective regularization parameters. It is widely known as image-classification deep learning model which has performed with more than 78% accuracy on ImageNet dataset. This model is combination of several ideas of computer vision experts over some years. It is explored by Szegedy, et al., (2016). A High level diagram is presented in Fig 5:



**Figure 5.** High-level diagram of InceptionV3 model (ref: Analytics-vidhya)

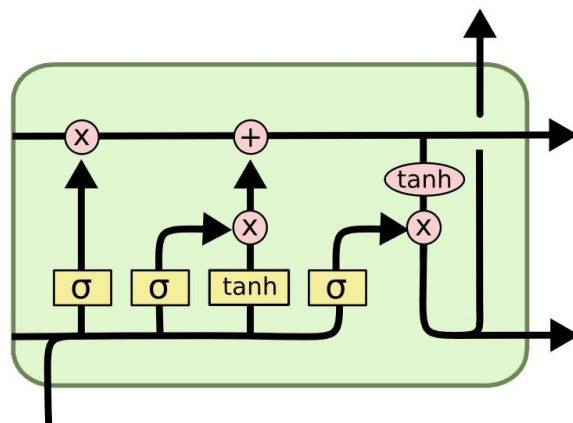
InceptionV3 model consists many symmetric and asymmetric building sections, which are made of convolution layers, average pooling layers, max pooling layers, con-cat layers, drop out layers and fully connected layers. Batch-norm is used at each layer to apply batch wise normalization, improve accuracy and faster the learning at each layer and it is applied at activation inputs. Classification loss is computed using Soft-max function. In our project, we will be using keras platform which already has pre-trained InceptionV3 model where model size is 92 MB and number of parameters will be 23,851,784.

### 3.2.2 GloVe Model

GloVe is known as Global vectors. Pennington et al, (2014) has explained this model very nicely. In recent approaches of learning word-embeddings, many approaches have been successful in achieving much fine-grained semantically and syntactic relationships via vector calculations. In this paper, he has explained the analysis of model properties which can introduce different type of regularities to appear in word-embeddings. GloVe is a global logbi-linear model which introduces the benefits of two important models together: one is global matrix factorization and other is local contextual window approach. GloVe model efficiency takes statistical information via model training on non-zero elements in a word to word cooccurrence 2-d vectors. It is not trained on fully sparse matrix in a big corpus of words. This model produces a meaningful sub-structure with more than 75% accuracy on some word analogy problems. As per GloVe website, GloVe is modeled on the basis of how next word will come after current word. We expect that the use of GloVe will improve image captioning better than just passing encoded words only. It has also performed better than some related models in entity recognition problems. In our project, we will be directly downloading GloVe vectors trained on Wikipedia and Gigaword 5. This data had 6 billion tokens and 0.4 million unique vocabs. In pre-trained GloVe model, there will 4 different type of word-embedding vectots 50dimensional, 100-dimensional, 200-dimensional, and 300-dimensional. In our study, we have considered 300-d word-embeddings to unlderstand the relationship between words. 300dimensional word-embeddings size is 1.013 GB.

### 3.3.3 LSTM Model

As explained in Wikipedia, Long-short-term-memory is an artificial recurrent neural network model was introduced by Hochreiter et al (1997). It has feedback connections unlike standard feedforward neural network models. It is not just limited to single-image processing but it can deal with complete sequence of image dataset or say video or speech dataset. As seen in below Fig 6, generic LSTM layer is made of an input gate, a cell, a forget gate and an output gate. The cell memorizes important values or information over random time intervals and all three gates regulate information flow into or out of a cell.



**Figure 6.** High-level diagram of LSTM networks (Ref: LSTM medium article)

LSTM is very useful for tasks like handwriting recognition, image captioning, speech recognition, traffic anomaly detection, video description etc. Also, LSTM networks are quite efficient for time series processing, predictions, or classification because there will be random duration between important information in time-series data. Initially, LSTM models were developed to get over with vanishing gradient problem but that can be dealt with traditional RNN models too. Relative in-sensitivity of time-duration between important values is a very big advantage of LSTM models over RNNs, or any other sequence processing methods like hidden Markov models etc.

### 3.3 Evaluation Metric

As explained by Papineni et.al, (2002), BLEU scores are known as Bilingual Evaluation Understudy. This score is a good metric to compare a candidate textual data to one or more reference textual descriptions. Mainly, it was introduced for machine translation evaluation, but it can be used for evaluation of other natural language processing problems also. BLEU scores can be calculated using text comparisons of 1-gram, 2-gram, 3-gram or any n-gram words. BLEU scores evaluation output will be always between 0 and 1. This score will represent the candidate text similarity with reference text data. An image captioning model will be considered good or bad depending on BLEU scores variation between 1 and 0. If score is 1 that model is excellent and if it is close to zero, that model is worse image captioning model. We have used `nlk.translate.bleu_score.corpus_bleu` function in Python to estimate BLEU scores, which is estimated using below formulas:

Combined BLEU scores:

$$BLEU_{n\_gram\_score} = BN * \exp(\sum_{n=1}^N w_n * \log(p_n))$$

where  $BN =$

$$\begin{cases} 1, & \text{if } len(predicted_{caption}) > len(smallest\ reference\ caption) \\ \exp\left(1 - \frac{len(smallest\ reference\ caption)}{len(predicted_{caption})}\right), & \text{else} \end{cases}$$

BN is known as brevity penalty and used to penalize small length predictions.

For sentence level precision estimation:

$$p_n = \frac{\text{number of } n_{gram} \text{ pairs from any of reference captions which belongs to predicted captions 'n}_{grams} \text{ pairs}}{\text{number of } n_{grams} \text{ in predicted caption}}$$

For corpus/document level precision estimation:

$$P_n = \frac{\text{number of } n_{gram} \text{ pairs from any of reference captions of that corpus which belongs to } n_{grams} \text{ pairs of predicted captions for that corpus}}{\text{total number of } n_{grams} \text{ pairs in predicted captions for all sentences of that corpus}}$$

## Chapter 4. Implementation

As described in introduction chapter, automatic image captioning means providing textual description of an image or the content observed in that image. We know that it's easy for human beings or even small kid to elaborate a scene in an image but it's not an easy job for a machine to describe the content of an image. In our project we are building a model which can describe an image for a given image. This chapter explain the complete implementation process from loading an image to predicting caption son new image. We have used Python language for complete implementation of image captioning model. In major python packages or platforms, we have used numpy for basic mathematic operations, nltk for text processing and opencv & keras for image processing related tasks. Both models cricket-captioning & Flickr-captioning, follow the same below mentioned implementation steps (after data-collection step):

## 4.1 Data collection

First step is to find suitable dataset to build an image captioning problem. As we are more interested towards sport specific captioning, we will be choosing cricket domain related images. We have downloaded around 1000 images from internet via python code. For cricket data, as we have very small dataset, we will use 800 images as train data and 200 images as test dataset. Although some of the images from cricket data will be filtered out due to nonsupported visuals like gif files. In our final cricket captioning model, we will have total 720 images, out of which 575 will be used for model training and remaining 145 will be used for model evaluation (test data). There are many open datasets like Flickr8k dataset, Flickr30k dataset & Microsoft COCO dataset which contains 180k images. We want to build an image captioning model using normal hardware and building model using large image dataset is not feasible on normal laptops. Therefore, we will be using Flickr8k dataset for this model development. This data can be downloaded from Kaggle platform without any cost. This dataset is a benchmark collection of 8000 images from different domains with 5 captions per image. All these captions provide major content info of any image. This dataset has pre-defined 6000 images as training data, 1000 images as development data and remaining 1000 images as test data. A sample of Flickr image captioning is provided below:



**Figure 7.** Sample image from Flickr8k data (Source: Flickr)

For above Fig 7, below are the 5 different captions provided:

*Caption1: a black dog is running after a white dog in the snow*

*Caption2: black dog chasing brown dog through snow*

*Caption3: two dogs chase each other across the snowy ground*

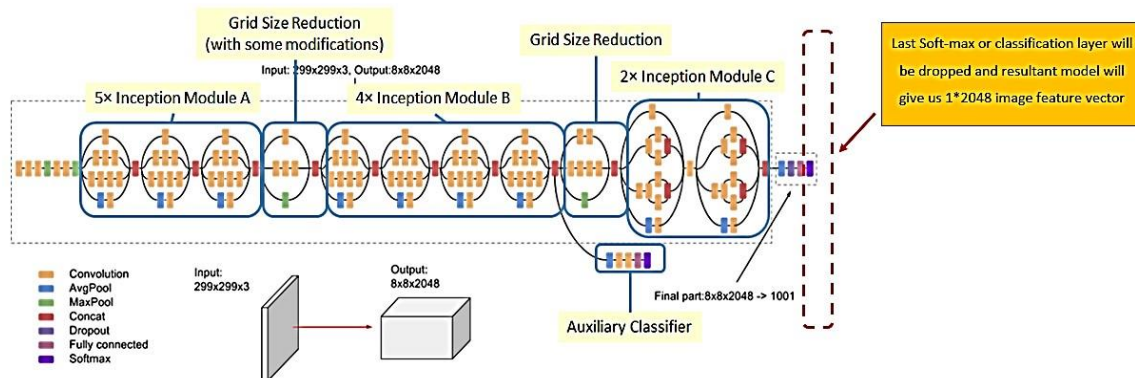
*Caption4: two dogs play together in the snow*

*Caption5: two dogs running through a low lying body of water*

## 4.2 Load image data & feature generation

We will load the images and use some standard deep neural network model to generate image feature vectors. In Keras pre-trained deep learning models, if we look at the performance analysis table of top-1 and top-5 object class prediction accuracy, then we see that InceptionV3 performs excellent among all above mentioned DL models. Therefore, we have used a pretrained InceptionV3 model to get image feature vectors. It is widely known as imageclassification deep learning model which has performed with more than 78% accuracy on ImageNet dataset. InceptionV3 is pre-trained on ImageNet dataset which has consists around 14 million images and 21 thousand object classes. Pre-trained InceptionV3 weights available on Keras had considered training on generic 1000 object classes only like people, car, cat, dog etc. We are using Keras platform to load this model and Keras provides these models with pretrained weights. Keras platform will download these weights (92 MB) and it will one-time activity. Once these weights are saved, it will help us to generate image features for any image. Once we load the InceptionV3 architecture and pre-trained weights, we can drop the

last layer of InceptionV3 model (softmax layer or classification layer) as out target is not to perform object classification. Our main priority is to generate detailed internal representation of an image. Fig 9. Provides a schematic representation of InceptionV3 model used for image feature extraction:



**Figure 8.** Schematic representation of InceptionV3 architecture for image captioning (ref: Analytics-vidhya)

With all above steps InceptionV3 architecture will extract an image feature vector given an image. Keras provides image reshaping option and pre-trained InceptionV3 has the image size requirements of 299\*299. Also, keras provide preprocess\_input function to perform preprocessing of an image as per pre-trained InceptionV3 model requirements. This complete process will generate image features of 1\*2048-dimensional vector.

### 4.3 Load image captions & text pre-processing

We know that image captioning is our main objective of project and each image of either cricket data or Flickr data has one or more captions. During model training captions will be out target variable which the model will learn to predict. In this dataset we need to perform at least minimal cleaning of captions. Each data has unique id for each image. It can be image name or image path etc. This will help us to align caption data and respective image. For any image id, we can find respective image captions. First, we will load all the captions and create a python dictionary where key will be image name and values will be the captions related to that image name or id. Once image captions loaded, text pre-processing will be performed to get optimized vocabulary or text corpus. Below text pre-processing steps are performed in our model:

- Converting all words in lower-case letters
- Dropping punctuations and numbers
- Dropping words with length lower than one

Above data cleaning steps will return an optimized and expressive vocabulary. If we get a smaller and expressive vocabulary that will help in faster and effective model training. After text pre-processing we will save the captions in below format:

*'captionstart' + caption + "captionend"*

We need to add captionstart and captionend in start and end of each caption because captionstart will notify the decoder to start captioning and captionend will notify the decoder to stop the captioning process for an image. It will be explained with an example later. For cricket train data, we have got 501 unique words out of which only 331 were more than once. Therefore, vocabulary size in cricket training model 331. Maximum length of any cricket training captions is 22. For Flickr train data, total unique words were 7579 out of which 2531

are repeated more than 5 times. Therefore, vocabulary size in cricket training model 2531. Maximum length of any Flickr training captions is 34.

#### 4.4 Loading pre-processed image and caption data

Cleaned descriptions of last step need to be encoded as numbers or each word should be assigned a number. This step will be helpful in upcoming sequence processing step. Keras provides a class named as Tokenizer which can learn the mapping for each word of a loaded description and assign a unique number to each word. Now we need to prepare the cleaned text data and image feature vectors as per model fitting requirements. We know that we cannot pass complete caption as target variable of image captioning model. We need to pass word-by-word. To perform that, we need to encode the captions. Each image caption will be divided into words. Model will be provided image feature vector and one word at a time and it will generate next word. Then first two words and image feature vectors will be given as inputs to generate next word. This will be the process of model training. Below example (Fig 9 and Table 9) provides an easy to understand explanation of the training data preparation:



**Figure 9.** A sample image of cricket data

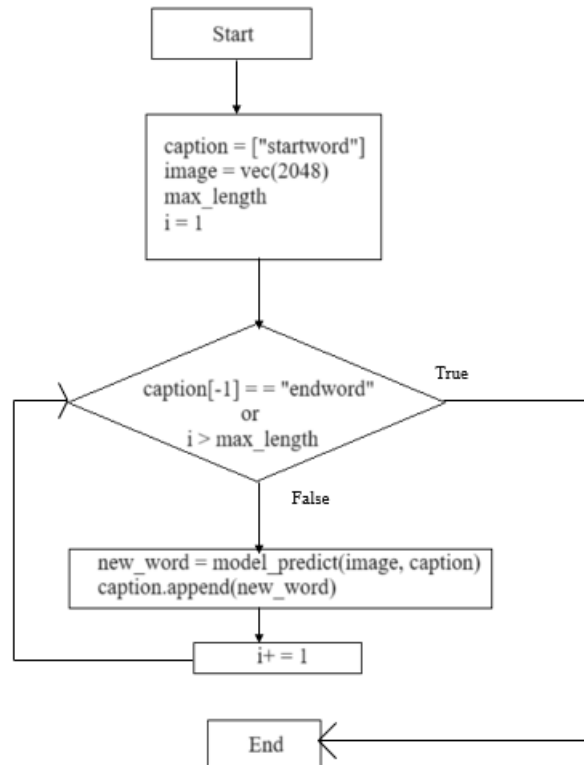
**Caption Generated:** A fielder is catching the ball

Table 1. Data preparation for image captioning training

S.N.	Image feature vector (Input1)	Word embeddings (Input2)	Output
1	Image_1	Captionstart	a
2	Image_1	captionstart a	fielder
3	Image_1	captionstart a fielder	is
4	Image_1	captionstart a fielder is	catching
5	Image_1	captionstart a fielder is catching	the
6	Image_1	captionstart a fielder is catching the	ball
7	Image_1	captionstart a fielder is catching the ball	captionend

#### 4.5 Defining and fitting the model

As explained in “Methodology” chapter in Fig 4, our model will be mainly described in three parts: image feature extractor, sequence processor and decoder. Image feature extractor contains the process of images feature vector extraction. Sequence processor will handle text inputs by generating word-embedding for each of word present in an image caption and it will be followed by a LSTM layer to understand the sequential relationships of textual data. Last part is decoder, which is basically merging of the fixed length output vectors of both image feature extractor and sequence processor steps. After merging, a dense Soft-max layer with be applied for next word prediction. Fig 8 provides a great flowchart to explain model fitting process:



**Figure 10.** Flowchart of model fitting steps

In this project, we haven't performed the model training by fitting all the train images at once because that will require be very time and resource consuming process normal laptops with 8 GB RAM. To deal with hard-ware or memory related issue, we have performed progressive loading to train the image captioning model. First of all, we need to define a generator function in python which is exactly built to deal with above mentioned issue. Generator is an iterator which resumes its functionality from the point where it left in last training phase. In our project we have passed 5 image per batch to keep the load on system at a time. When calling a data generator function, we need to provide the details like prepared image feature vectors and caption data, tokenizer and maximum length of captions observed in train data. In each training batch of 5 images, number of training samples will be dependent on its caption length and words used in it. A training sample for an image and caption is provided in Table 1. If we look at the image and caption provided at Table 1 that will result into 7 training samples. Therefore, number of training samples per batch won't be constant here. All above steps will pretty much solve system RAM or major hardware requirement related problems. We can save the trained model at each epoch and finalize the best model by checking loss on development dataset. In next step, we will be selecting best epoch-fitted model using the predicted captions on test images.

#### 4.6 Finalizing the model

As mentioned in last step, we have saved fitted or trained models at each epoch. Now we need to find the best-fitted model using any unseen dataset. For Flickr dataset, we will used development dataset, which has around 1000 images and for cricket data, we will use test data which has 145 images. Best epoch-fitted model selection will be based on BLEU-1 & BLEU2 scores. Results or plots related to model selection will be presented in next chapter. Once we finalize the model, we will more for model evaluation on test dataset.



## 4.7 Model evaluation

Once the prediction on test images are made using above step, we can perform captioning evaluation on test data. In this step, we will take the predicted captions of last step and check the caption prediction performance using BLEU scores mainly cumulative unigram (BLEU-1) & cumulative bi-gram (BLEU-2) scores. Details regarding BLEU scores are presented in “Methodology” chapter and mathematical details in next chapter. The way we have trained or fitted the image captioning model, it will predict the image caption is similar way:

*‘captionstart’ + caption + “captionend”*

Once we get the prediction done for each of test image, we can go for BLEU score estimation. BLEU scores evaluation output will be always between 0 and 1. This score will represent the candidate text similarity with reference text data. An image captioning model will be considered good or bad depending on BLEU scores variation between 1 and 0. If score is 1 that model is excellent and if it is close to zero, that model is worse image captioning model. We will calculate the BLEU score for predicted caption and each of reference caption for each throughout complete test data.

## 4.8 Caption generation for test or new images

Once an image captioning model is trained and best fitted model is selected on basis of BLEU1 & BLEU-2 scores, we can use that to predict captions for new images. As we haven’t seen test images yet, we can predict the captions for test images. We will load the tokenizer saved during model training. It will reduce the task of loading complete caption data again. We need to define max\_length of captioning also which can be same as found during model training. Once a test image is loaded, we can generate the features using pre-trained InceptionV3 model. Now, image feature vector and caption start word “captionstart” and max\_length will be passed to the trained model and to get the probability of next word. Word with maximum probability will be the next word in sequence and this process will be repeated until we get “captionend” word. Then, we can remove start and end words (“captionstart” and “captionend”) which will give us predicted caption for test image. Once we complete the caption predictions for all test images, we will move on model evaluation step.

More insights related to image captioning findings are provided in next chapter “Results and Analysis”.

# Chapter 5. Results and Analysis

In this chapter, we will present all the major findings of image captioning model. First, we will elaborate on different BLEU-scores’ estimation via examples. Then, we will explore the results of image captioning model built using cricket data, Flickr data and comparisons. Generally, there are multiple type of BLEU score estimations. We have presented some of these score estimations via examples:

## 5.1 Image captioning model summary

Below we are presenting the summary of model used for image captioning using cricket or Flickr data. It provides the information of each layer, inputs & outputs etc. This model structure is same for both cricket and Flickr trained models

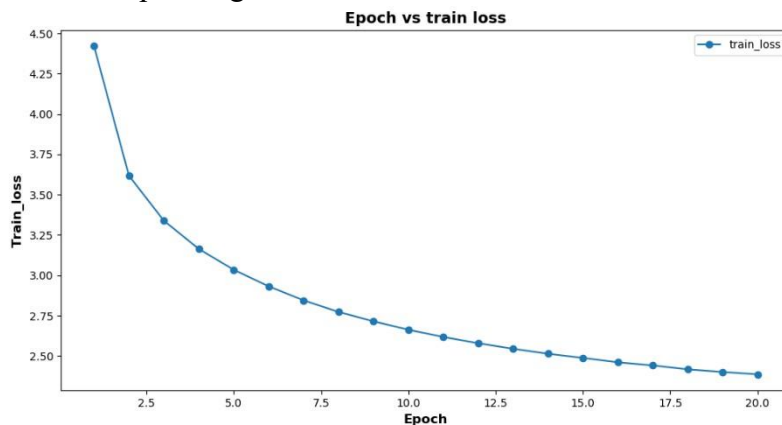
Table 2. Image captioning model summary

Layer (type)	Output Shape	Param #	Connected to
input_21 (InputLayer)	(None, 22)	0	
input_22 (InputLayer)	(None, 2048)	0	
embedding_11 (Embedding)	(None, 22, 300)	99300	input_21[0][0]
dropout_22 (Dropout)	(None, 2048)	0	input_22[0][0]
dropout_21 (Dropout)	(None, 22, 300)	0	embedding_11[0][0]
dense_31 (Dense)	(None, 300)	614700	dropout_22[0][0]
lstm_11 (LSTM)	(None, 300)	721200	dropout_21[0][0]
add_11 (Add)	(None, 300)	0	dense_31[0][0] lstm_11[0][0]
dense_32 (Dense)	(None, 300)	90300	add_11[0][0]
dense_33 (Dense)	(None, 331)	99631	dense_32[0][0]

Total params: 1,625,131  
 Trainable params: 1,525,831  
 Non-trainable params: 99,300

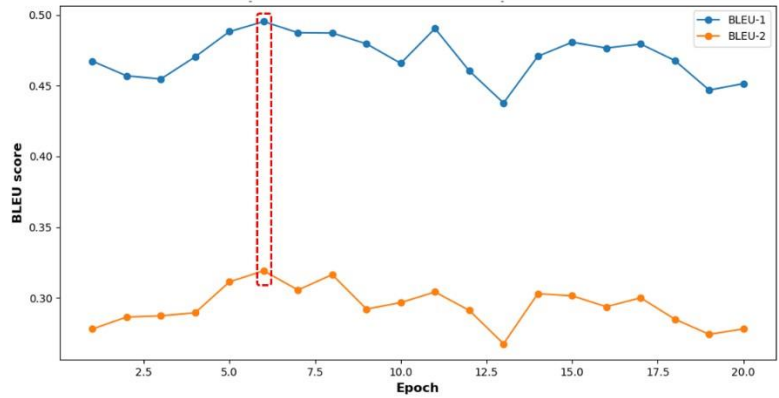
## 5.2 Case Study 1 – Flickr image captioning model:

In this section, we will present the results of image captioning model obtained trained on Flickr cricket data. Here we have trained Flickr image captioning model for 20 epochs. Fig 14. Presents epoch vs train loss plot. In this plot, we can clearly see that train loss is constantly decreasing with increase in epochs. Although, this plot doesn't give us information about the best epoch-fitted cricket captioning model.



**Figure 11.** Epoch vs train loss in image captioning model

To finalize the best epoch-fitted model, we need to check checking BLEU-scores on Flickr test data. We have predicted cumulative BLEU-scores on test data using trained Flickr captioning models starting from 1<sup>st</sup> epoch to 20<sup>th</sup> epoch. Fig 13 presents the plot of cumulative BLEU scores (BLEU-1 & BLEU-2) on cricket testing data. We found that 6<sup>th</sup> epoch-trained model is giving best uni-gram & bi-gram cumulative BLEU scores. Hence, we will use this finalized (6<sup>th</sup> epoch) model to test on some images from Flickr test data and on some cricket images.



**Figure 12.** Epoch vs train loss in image captioning model



After finalizing the best fitted cricket trained image captioning model (6<sup>th</sup> epoch), we have checked the cumulative BLEU scores on testing cricket data and below are results we got:

BLEU-1: 0.467102

BLEU-2: 0.285009

These scores are telling us that on an average our prediction are ~47% similar to test image reference captions (Flickr) if we go with uni-gram tokens, and ~23% similar to test image reference captions (Flickr) if we go with bi-gram tokens. Although this score looks average, but these are better than cricket testing BLEU scores.

Below we will be presenting some of the images from Flickr test data and putting the respective BLEU-1 & BLEU-2 scores:

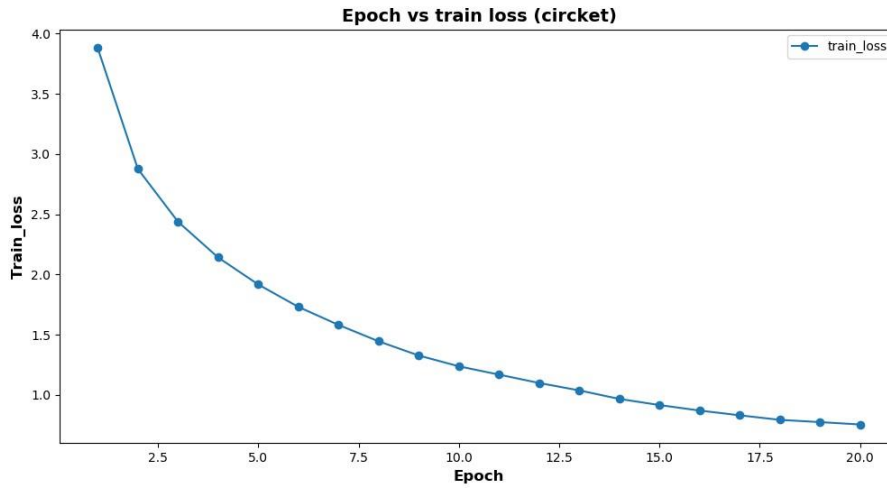
Image-1	Reference Captions	Predicted caption
	['the dogs are in the snow in front of fence', 'the dogs play on the snow', 'two brown dogs playfully fight in the snow', 'two brown dogs wrestle in the snow', 'two dogs playing in the snow']	['two dogs are running through the grass'] BLEU scoring:: BLEU-1: 0.571429 BLEU-2: 0.436436
	['the dogs are in the snow in front of fence', 'the dogs play on the snow', 'two brown dogs playfully fight in the snow', 'two brown dogs wrestle in the snow', 'two dogs playing in the snow']	['two dogs are running through the grass'] BLEU scoring:: BLEU-1: 0.571429 BLEU-2: 0.436436
Image-3	Reference Captions	Predicted caption

	<p>['girl with dark brown hair and eyes in blue scarf is standing next to girl in fur edged coat', 'an asian boy and an asian girl are smiling in crowd of people', 'the girls were in the crowd', 'two dark haired girls are in crowd', 'two girls are looking past each other in different directions while standing in crowd']</p>	<p>['man and woman pose for picture'] BLEU scoring:: BLEU-1: 0.166667 BLEU-2: 0.000000</p>
Image-4	Reference Captions	Predicted caption
	<p>['player from the white and green highschool team dribbles down court defended by player from the other team', 'four basketball players in action', 'four men playing basketball two from each team', 'two boys in green and white uniforms play basketball with two boys in blue and white uniforms', 'young men playing basketball in competition']</p>	<p>['basketball player in white uniform is playing basketball'] BLEU scoring:: BLEU-1: 0.625000 BLEU-2: 0.298807</p>
Image-5	Reference Captions	Predicted caption
	<p>['man helps another man tie red ribbon onto his arm', 'man helps tie red ribbon around another man right arm during street parade', 'man is tying red arm band around another mans arm in the street', 'one man helps another attach red ribbon to his forearm in the midst of large group of people', 'two men stand together one is putting something red on his arm']</p>	<p>['man in black shirt and tie is standing in front of crowd of people'] BLEU scoring:: BLEU-1: 0.500000 BLEU-2: 0.196116</p>

Flickr trained model is giving quite good prediction from image-1 to image-5. Now, we will check the prediction of Flickr trained model on some cricket or other sports related images.

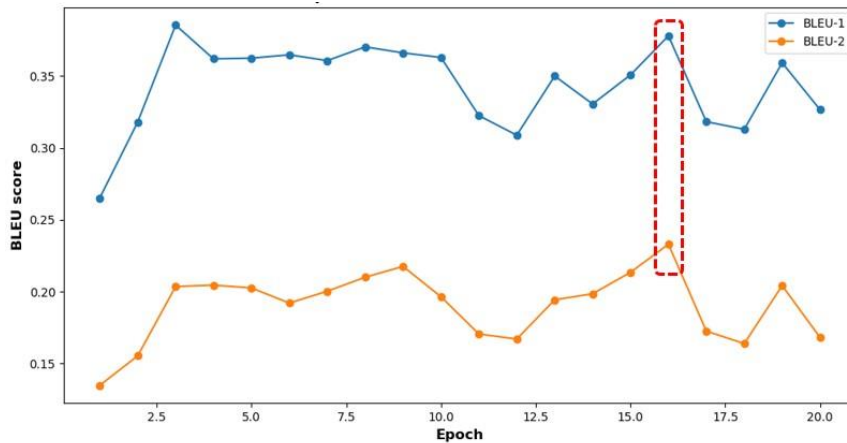
### 5.3 Case Study 2 – Cricket image captioning model

First, we will present the results of image captioning model obtained using cricket data. Here we have train data to fir the cricket captioning model for 20 epochs. Fig 12. Presents epoch vs train loss plot. In this plot, we can clearly see that train loss is constantly decreasing with increase in epochs. Although, this plot doesn't give us information about the best epoch-fitted cricket captioning model



**Figure 13.** Epoch vs train loss in cricket captioning model

To finalize the best epoch-fitted model, we need to check checking BLEU-scores on testing cricket data. We have predicted cumulative BLEU-scores on test data using trained cricket captioning models starting from 1<sup>st</sup> epoch to 20<sup>th</sup> epoch. Fig 13 presents the plot of cumulative BLEU scores (BLEU-1 & BLEU-2) on cricket testing data. We found that 16<sup>th</sup> epoch-trained model is giving best uni-gram & bi-gram cumulative BLEU scores. Hence, we will use this finalized (16<sup>th</sup> epoch) model to test on some cricket images from test data and on some out of domain images.



**Figure 14.** Trained models at each epoch vs BLEU scores on cricket test data

After finalizing the best fitted cricket trained image captioning model (16<sup>th</sup> epoch), we have checked the cumulative BLEU scores on testing cricket data and below are results we got:


BLEU-1: 0.377960

BLEU-2: 0.232975


These scores are telling us that on an average our prediction are ~38% similar to test images (cricket) if we go with uni-gram tokens, and ~23% similar to test images (cricket) if we go with bi-gram tokens. Although this score looks quite low, but looking at the smaller training data, we can't expect more.

Below we will be presenting some of the test images from cricket data and putting the respective BLEU-1 & BLEU-2 scores:


Image-1	Reference Captions	Predicted caption
---------	--------------------	-------------------

	['batsman jumped to hit the ball', 'batsman slightly jumped up to hit the ball']	['batsman is trying to hit the ball with bat'] BLEU scoring:: BLEU-1: 0.555556 BLEU-2: 0.456435
---	--	--


In image-1, we can clearly see that our prediction is quite similar to reference caption.

Image-2	Reference Captions	Predicted caption
	['an umpire is standing with his right index finger raised']	['an umpire with white shirt and white hat is standing next to wicket'] BLEU scoring:: BLEU-1: 0.384615 BLEU-2: 0.253185

In image-2, model is able to detect umpire but color and wicket part is predicted wrong. Also, we haven't more than 5 cartoon images in training data. Still, model is able to identify umpire just by posture identification.


Image-3	Reference Captions	Predicted caption
	['the batsman practicing on the pitch', 'batsman practicing how to defend the ball', 'batsman in white dress trying to defend the ball', 'batsman in white dress is on the pitch', 'batsman practicing before the match']	['two batsman are on the cricket field'] BLEU scoring:: BLEU-1: 0.428571 BLEU-2: 0.267261

In image-3, model is detecting two batsman instead of one.


Image-4	Reference Captions	Predicted caption
	['the left handed bowler has bent over to bowl spin ball']	['fielder trying to catch the ball with right hand'] BLEU scoring:: BLEU-1: 0.266912 BLEU-2: 0.000000

In image-4, model is confused by bowler's action and predicting him as a fielder. These things can be corrected with large amount of data.


Image-5	Reference Captions	Predicted caption
---------	--------------------	-------------------

	['the fielder caught the ball with two hands', 'he is one of the fielder in cricket', 'the fielder caught the white ball with two hands']	['person in the field caught the ball'] BLEU scoring:: BLEU-1: 0.619198 BLEU-2: 0.422993
---	---	---


In image-5, we can clearly see that our prediction is quite similar to reference caption. Prediction looks better than one of the reference too.

Image-6	Reference Captions	Predicted caption
	['batsman is wearing hat']	['batsman is trying to hit the ball with bat'] BLEU scoring:: BLEU-1: 0.222222 BLEU-2: 0.166667

In image-6, even though BELU score is lower but model prediction is far better than reference provided.


Image-7	Reference Captions	Predicted caption
	['an umpire is walking away from wicket', 'person in white shirt is walking on field', 'person in white shirt is gesturing with his right hand', 'an umpire in white hat is walking in cricket field']	['an umpire with white hat on his belt belt belt in'] BLEU scoring:: BLEU-1: 0.727273 BLEU-2: 0.381385

In image-7, model is able to predict most of the context but sentence is not completed.


Image-8	Reference Captions	Predicted caption
	['person with blue cap is wicket keeper', 'person holding the bat is batsman']	['batsman bowled out'] BLEU scoring:: BLEU-1: 0.122626 BLEU-2: 0.000000

In image-8, model is able to predict that someone as thrown a ball on wickets and wickets are misplaced. Even though it looks like a runout here but model is predicting more scene context than given in reference captions.

Image-9	Reference Captions	Predicted caption
---------	--------------------	-------------------



	<p>['person in green dress is wicket keeper', 'wicket keeper stumping the wickets']</p>	<p>['batsman touches the wicket with his bat'] BLEU scoring:: BLEU-1: 0.285714 BLEU-2: 0.000000</p>
---	---	---

In image-9, model is trying to relate all the scene objects which are not completely described in reference captions.

Image-10	Reference Captions	Predicted caption
	<p>['umpire in red signalling wide']</p>	<p>['an umpire is carrying white jacket'] BLEU scoring:: BLEU-1: 0.166667 BLEU-2: 0.000000</p>

In image-10, prediction is slightly wrong here.

Predictions on images from different domains using cricket captioning model:

Image	Predicted caption
	<p>batsman is trying to take run</p>
	<p>person in the field is demanding to catch the ball</p>





two players are on the cricket player

In above three images, we know that prediction are not good at all. Cricket captioning model is not trained on the domains of above images or generic images. Therefore, this model will not work on images apart from cricket sport.

## Chapter 6. Conclusion & Future Scope

In this study we have worked on automatic image captioning model. Automatic caption generation means providing textual description of an image or the content observed in that image. It has been a difficult problem in both computer vision and natural language processing. In recent years, it has been a topic of interest due to its importance in real-world applications such as self-driving cars, visual assistants for blind/needily individuals, and the improvement of malicious activity detection, as well as because it involves two very important manufactured fields: computer vision and natural language processing.. We know that it's quite easy and simple for human beings to provided textual description of an image, but main challenge is to make machines learn this task. Therefore, we have worked on image captioning models starting from loading the image and textual data to ending with caption prediction on new images. We have downloaded around 1000 cricket images via python script and put the captions manually on an average 2 captions per image. Apart from it, we have used a standard dataset: Flickr8k also. Flickr8k has 6000 train images, 1000 development images & 1000 test images. Flickr8k data has 5 captions per image for all train, development & test images. All the coding parts were completed in Python language and some important packages like Keras & NLTK to perform computer vision and NLP related tasks. We can briefly summarize the major tasks completed in this project: First we need to get image feature vectors to represent an image and we have used pre-trained InceptionV3 to get it done. Next image captions were pre-processed and encoded as per model training requirements.

Here captions are encoded with 'captionstart' word in starting and 'captionend' word in ending of each caption. Both will have the role in kick-off captioning process and stop it respectively. Once text data is pre-processed, we get word-embedding for each word using pre-trained GloVe vectors. GloVe vectors introduces a better relationship between two neighboring words of a caption depending on their use in real-life. Next words will be passed via LSTM model to get the sequential processing done. Now, we won't be taking complete caption as target variable here. We need to go word-by-word prediction approach. We will be having two inputs in the model, one is image feature vector and other is partial caption, which will help us to get net word in caption-sequence. Now, the output of sequential processing and image factor will be merged using dense Soft-max layer which will predict next word. This predicted word will be added in the text sequence for next step or next word's prediction. Once training samples with two input and a target variable prepared, we will be fitting the model with progressive loading approach. We don't fit all the training samples at once to avoid system-memory error and to

save good amount to training time. In progressive loading we have passed the training samples prepared using 5 images and respective captions per batch. It resolves the requirement of big system hardware requirements and is suitable to work on a normal PC/laptop. We have performed the model training up to 20 epochs and saved the results of each epoch. Next, we will test the saved models of each epoch on development dataset to finalize the best trained model using BLEU scores on test/development datasets. Once it's finalized, we'll move on to predict the captions on test data. Then we'll evaluate the predicted captions using BLEU score. BLEU score match the similarity between predicted caption and one of the reference captions. We have two type of BLEU scores for model selection or evaluation purposes: uni-gram cumulative BLEU score (BLEU-1) & bi-gram cumulative BLEU score (BLEU-2). For cricketcaptioning model we have finalized 16th epoch-fitted model as best performing model and got BLEU-1 score as 0.38 & BLEU-2 score as 0.23. Then, we performed testing of this cricket captioning model on some cricket test images where we are getting quite satisfactory caption predictions. Although this model is not able to give good captions on different domain fields. Then, we performed the same steps for Flickr-captioning model and found 6th epoch gives best model fitting. Flickr model resulted BLEU-1 as 0.47 and BLEU-2 as 0.29 on test dataset. Flickr gives quite satisfactory predictions on multiple different domains compare to cricket captioning model. In future, some of points which can helps our cricket captioning model are put at least 10-15 words captions for each image, train data should have more than 10000 images with good amount of cricket scene variations etc. In Future, we can explore image captioning with large cricket or any other sports data, hyper-parameter tuning, large CNN models like Resnet152, InceptionResNetV2 or NASNetLarge etc. to detect image feature vectors. Further, we can explore on use of attention-based module too. In future, we can explore the integration of image captioning with other domain also like assistance to needy/blind people will require perspective scene to textual description and then textual description to audio generation.

## References

- Agrawal, H.; Desai, K.; Chen, X.; Jain, R.; Batra, D.; Parikh, D.; Lee, S.; Anderson, P. Nocaps: Novel object captioning at scale. arXiv 2018, arXiv:1812.08658.
- Aneja, J., Deshpande, A., & Schwing, A. G. (2018). Convolutional image captioning. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 5561-5570).
- Barnard, K., Duygulu, P., Forsyth, D., De Freitas, N., Blei, D.M. and Jordan, M.I., 2003. Matching words and pictures. The Journal of Machine Learning Research, 3, pp.1107-1135
- Chen, X., & Lawrence Zitnick, C. (2015). Mind's eye: A recurrent visual representation for image caption generation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2422-2431).
- Chen, Y.C., Li, L., Yu, L., El Kholi, A., Ahmed, F., Gan, Z., Cheng, Y. and Liu, J., 2020, August. Uniter: Universal image-text representation learning. In European conference on computer vision (pp. 104-120). Cham: Springer International Publishing
- Elliott, D. and Kádár, A., 2017. Imagination improves multimodal translation. arXiv preprint arXiv:1705.04350 Elliott, D. and Kádár, A., 2017. Imagination improves multimodal translation. arXiv preprint arXiv:1705.04350
- Elliott, D., & Keller, F. (2013, October). Image description using visual dependency representations. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (pp. 1292-1302).
- Elliott, D., Frank, S., Barrault, L., Bougares, F. and Specia, L., 2017. Findings of the second shared task on multimodal machine translation and multilingual image description. arXiv preprint arXiv:1710.07177
- Elliott, D., Frank, S., Sima'an, K. and Specia, L., 2016. Multi30k: Multilingual english-german image descriptions. arXiv preprint arXiv:1605.00459

- Fan, C.; Crandall, D.J. Deep Diary: Automatic caption generation for lifelogging image streams. In Proceedings of the European Conference on Computer Vision, Amsterdam, the Netherlands, 11–14 October 2016; pp. 459–473. [CrossRef]
- Fang, H., Gupta, S., Iandola, F., Srivastava, R. K., Deng, L., Dollár, P., ... & Lawrence Zitnick, C. (2015). From captions to visual concepts and back. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1473-1482).
- Farhadi, A., Hejrati, M., Sadeghi, M. A., Young, P., Rashtchian, C., Hockenmaier, J., & Forsyth, D. (2010, September). Every picture tells a story: Generating sentences from images. In European conference on computer vision (pp. 15-29). Springer, Berlin, Heidelberg.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-1780.
- Jewitt, C., Kress, G., Ogborn, J. and Tsatsarelis, C., 2001. Exploring learning through visual, actional and linguistic communication: The multimodal environment of a science classroom. *Educational Review*, 53(1), pp.5-18
- Karpathy, A., & Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3128-3137).
- Kilickaya, M., Erdem, A., Ikizler-Cinbis, N., & Erdem, E. (2016). Re-evaluating automatic metrics for image captioning. arXiv preprint arXiv:1612.07600.
- Kinghorn, P.; Zhang, L.; Shao, L. A region-based image caption generator with refined descriptions. *Neurocomputing* 2018, 272, 416–424. [CrossRef]
- Kiros, R., Salakhutdinov, R. and Zemel, R., 2014, June. Multimodal neural language models. In International conference on machine learning (pp. 595-603). PMLR
- Kiros, R., Salakhutdinov, R. and Zemel, R.S., 2014. Unifying visual-semantic embeddings with multimodal neural language models. arXiv preprint arXiv:1411.2539
- Kiros, R., Salakhutdinov, R., & Zemel, R. (2014, January). Multimodal neural language models. In International conference on machine learning (pp. 595-603).
- KOUSTUBH. ResNet, AlexNet, VGGNet, Inception: Understanding Various Architectures of Convolutional Networks. Available online: <https://cv-tricks.com/cnn/understand-resnet-alexnet-vgginception/> (accessed on 24 May 2019).
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).
- Li, S., Kulkarni, G., Berg, T., Berg, A., & Choi, Y. (2011, June). Composing simple image descriptions using web-scale n-grams. In Proceedings of the Fifteenth Conference on Computational Natural Language Learning (pp. 220-228).
- Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common objects in context. In Proceedings of the Computer Vision—ECCV 2014, Zurich, Switzerland, 6–12 September 2014; Springer International Publishing: Cham, Switzerland, 2014; pp. 740–755. [CrossRef]
- Mao, J., Xu, W., Yang, Y., Wang, J. and Yuille, A.L., 2014. Explain images with multimodal recurrent neural networks. arXiv preprint arXiv:1410.1090
- Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z. and Yuille, A., 2014. Deep captioning with multimodal recurrent neural networks (m-mn). arXiv preprint arXiv:1412.6632
- Martinec, R. and Salway, A., 2005. A system for image–text relations in new (and old) media. *Visual communication*, 4(3), pp.337-371
- Mishra, A.; Liwicki, M. Using deep object features for image descriptions. arXiv 2019, arXiv:1902.09969.
- Mithun, N.C., Li, J., Metze, F. and Roy-Chowdhury, A.K., 2018, June. Learning joint embedding with multimodal cues for cross-modal video-text retrieval. In Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval (pp. 19-27)
- Mnih, V., Heess, N., & Graves, A. (2014). Recurrent models of visual attention. In Advances in neural information processing systems (pp. 2204-2212).

Ordonez, V.; Han, X.; Kuznetsova, P.; Kulkarni, G.; Mitchell, M.; Yamaguchi, K.; Stratos, K.; Goyal, A.; Dodge, J.; Mensch, A.; et al. Large scale retrieval and generation of image descriptions. *Int. J. Comput. Vis.* 2016, 119, 46–59. [CrossRef]

Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002, July). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics* (pp. 311-318).

Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532-1543).

Plummer, B.A.; Wang, L.; Cervantes, C.M.; Caicedo, J.C.; Hockenmaier, J.; Lazebnik, S. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015*; pp. 2641–2649. [CrossRef]

Qi, J. and Peng, Y., 2018, July. Cross-modal Bidirectional Translation via Reinforcement Learning. In *IJCAI* (pp. 2630-2636)

Sankey, M., Birch, D. and Gardiner, M.W., 2010. Engaging students through multimodal learning environments: The journey continues. *Proceedings of the 27th Australasian Society for Computers in Learning in Tertiary Education*, pp.852-863

Schmidhuber, J., & Hochreiter, S. (1997). Long short-term memory. *Neural Comput*, 9(8), 1735-1780.  
Yao, T., Pan, Y., Li, Y., Qiu, Z., & Mei, T. (2017). Boosting image captioning with attributes. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 4894-4902).

Specia, L., Frank, S., Sima'An, K. and Elliott, D., 2016, August. A shared task on multimodal machine translation and cross-lingual image description. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers* (pp. 543-553)

Staniūtė, R., & Šešok, D. (2019). A systematic literature review on image captioning. *Applied Sciences*, 9(10), 2024.

Summaira, J., Li, X., Shoib, A.M., Li, S. and Abdul, J., 2021. Recent advances and trends in multimodal deep learning: a review. *arXiv preprint arXiv:2105.11087*

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2818-2826).

Tang, C., Srivastava, N., & Salakhutdinov, R. R. (2014). Learning generative models with visual attention. In *Advances in Neural Information Processing Systems* (pp. 1808-1816).

Tariq, A.; Foroosh, H. A context-driven extractive framework for generating realistic image descriptions. *IEEE Trans. Image Process.* 2017, 26, 619–632. [CrossRef]

Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3156-3164).

Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2016). Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *IEEE transactions on pattern analysis and machine intelligence*, 39(4), 652663.

Zhang, Y., Jiang, H., Miura, Y., Manning, C.D. and Langlotz, C.P., 2022, December. Contrastive learning of medical visual representations from paired images and text. In *Machine Learning for Healthcare Conference* (pp. 2-25). PMLR

Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O. and Shechtman, E., 2017. Toward multimodal image-to-image translation. *Advances in neural information processing systems*, 30