



Technological University Dublin
ARROW@TU Dublin

Dissertations

School of Computer Sciences

2021

Adequately Generating Captions for an Image Using Adaptive and Global Attention Mechanisms.

Shravan Kumar Talanki Venkatarathanaiahsetty
Technological University Dublin

Follow this and additional works at: <https://arrow.tudublin.ie/scschcomdis>

 Part of the [Computer Engineering Commons](#)

Recommended Citation

Venkatarathanaiahsetty, S. K. T. (2021). *Adequately generating captions for an image using adaptive and global attention mechan.isms*. Dissertation. Dublin: Technological University Dublin. doi: [https:10.21427/wv3a-md49](https://doi.org/10.21427/wv3a-md49)

This Dissertation is brought to you for free and open access by the School of Computer Sciences at ARROW@TU Dublin. It has been accepted for inclusion in Dissertations by an authorized administrator of ARROW@TU Dublin. For more information, please contact arrow.admin@tudublin.ie, aisling.coyne@tudublin.ie.



This work is licensed under a [Creative Commons Attribution-Noncommercial-Share Alike 3.0 License](#)



**Adequately generating captions for an image using
Adaptive and Global attention mechanisms**



Shravan Kumar Talanki Venkatarathanaiahsetty

Technological University, Dublin

A dissertation submitted in partial fulfilment of the requirements of
Technological University Dublin for the degree of
M.Sc. in Computer Science (Data Science)

2021

DECLARATION

I certify that this dissertation which I now submit for examination for the award of MSc in Computing (Data Science), is entirely my work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

This dissertation was prepared according to the regulations for postgraduate study of the Technological University Dublin and has not been submitted in whole or part for an award in any other Institute or University.

The work reported on in this dissertation conforms to the principles and requirements of the Institute's guidelines for ethics in research.

Shraavan Kumar, T.U

Signed:

Date: **05 January 2021**

ABSTRACT

Generating description to images is a recent surge and with latest developments in the field of Artificial Intelligence, it can be one of the prominent applications to bridge the gap between Computer vision and Natural language processing fields. In terms of the learning curve, Deep learning has become the main backbone in driving many new applications. Image Captioning is one such application where the usage of Deep learning methods enhanced the performance of the captioning accuracy. The introduction of the Encoder-Decoder framework was a breakthrough in Image captioning. But as the sequences got longer the performance of captions was affected. To overcome this the usage of the attention mechanism as an extension to the Encoder-Decoder framework became an upward trend. Where an Attention mechanism generates a context vector having calculated information of pixels and using this information the decoder focuses on a particular region of an image and generates caption. Researchers proposed various attention mechanisms to generate a context vector having calculated information of image pixels. [Luong et al. \(2015\)](#) are one such who proposed a Global attention mechanism that makes a decoder look at the calculated pixels of the image at each time step while generating the caption. Similarly, an attention mechanism named Adaptive attention was proposed by [Lu et al. \(2017\)](#) which allows the decoder to decide whether the calculated pixels of an image need to be focused at each time step or needs to concentrate on a language model.

This research proposes a comparative study of these two attention mechanisms in the generation of captions for images using the Flickr30k dataset. A deep Residual Network with 152 layers (ResNet-152) is used as an encoder and an LSTM is used as the decoder. An evaluation of the model is performed using BLEU, METEOR, ROUGE, CIDEr metrics and results show the usage of Adaptive attention over Global attention would yield better metric scores.

Keywords: *Image Captioning, Deep learning, Attention Mechanism, Global Attention, Adaptive attention, ResNet, LSTM*

ACKNOWLEDGEMENTS

I would like to express my sincere thanks to my supervisor, Giancarlo Salton for his valuable help and guidance. I am thankful for the encouragement provided by him all these days in completing my thesis. He has great patience and it was easy to work with him.

I want to thank all the professors at TU Dublin who have helped me create a good base so that I can approach thesis in the best possible shape.

I would like to thank Prof. John Gilligan and Dr Luca Longo for their advice and help to narrow down my field of interest.

Finally, I would like to thank my parents for their support and confidence in me which helped to maintain a positive attitude and complete my thesis.

TABLE OF CONTENTS

DECLARATION	II
ABSTRACT	III
ACKNOWLEDGEMENTS	IV
TABLE OF FIGURES	VII
TABLE OF TABLES	VIII
1. INTRODUCTION.....	1
1.1 BACKGROUND	2
1.2 RESEARCH PROBLEM	3
1.3 RESEARCH OBJECTIVES	4
1.4 RESEARCH METHODOLOGIES	4
1.5 SCOPE AND LIMITATIONS	5
1.6 DOCUMENT OUTLINE	5
2. LITERATURE REVIEW	7
2.1. IMAGE CAPTIONING.....	7
2.1.1 <i>Template-based approaches</i>	7
2.1.2 <i>Retrieval based approaches</i>	8
2.1.3 <i>Neural based approaches</i>	9
2.2. DEVELOPMENTS IN CNN.....	10
2.3. DEVELOPMENTS IN LSTM.....	14
2.3.1 <i>Attention mechanism</i>	17
2.3.2 <i>Usage of attention mechanism as an extension to Decoder for achieving Image Captioning task</i>	21
2.4. SUMMARY, LIMITATIONS AND GAPS IN THE LITERATURE	25
3. DESIGN AND METHODOLOGY	26
3.1 BUSINESS UNDERSTANDING	26

3.2	HYPOTHESIS	27
3.3	UNDERSTANDING OF DATA	27
3.4	RESNET-152 AS ENCODER	28
3.5	ATTENTION MECHANISM FOR IMAGE CAPTIONING	31
3.6	LSTM AS DECODER	32
3.7	EVALUATION METRICS	33
4.	IMPLEMENTATION AND RESULTS	35
4.1	DATA PRE-PROCESSING	35
4.2	RESNET 152 AS ENCODER	36
4.3	GLOBAL ATTENTION AND ADAPTIVE ATTENTION	38
4.4	DECODER (LSTM) WITH AN ATTENTION MECHANISM	40
4.5	OVERALL ARCHITECTURE AND TRAINING PROCESS	42
4.6	RESULTS.....	44
4.6.1	<i>Quantitative results</i>	44
4.6.2	<i>Qualitative results</i>	46
5.	EVALUATION AND DISCUSSION	51
5.1	EVALUATION OF RESULTS	51
5.2	STRENGTHS OF RESULT	52
5.3	LIMITATIONS OF RESULT	52
6.	CONCLUSION AND FUTURE WORK	54
6.1	OVERVIEW OF RESEARCH AND EXPERIMENT	54
6.2	CONTRIBUTIONS AND IMPACT	55
6.3	FUTURE WORK & RECOMMENDATIONS	55
	BIBLIOGRAPHY	57
	APPENDIX A.....	62
	BAD CAPTIONS GENERATED BY THE GLOBAL ATTENTION MODEL	62
	BAD CAPTIONS GENERATED BY THE ADAPTIVE ATTENTION MODEL.....	63

TABLE OF FIGURES

FIGURE 2.1: ALEXNET ARCHITECTURE DEMONSTRATING FIVE CONVOLUTION AND THREE FULLY CONNECTED LAYERS (KRIZHEVSKY, SUTSKEVER & HINTON, 2012).	11
FIGURE 2.2: INCEPTION BLOCK ARCHITECTURE USED IN GOOGLNET SHOWING THE SPLIT, TRANSFORM AND MERGE CONCEPT (SZEGEDY ET AL., 2015).	12
FIGURE 2.3: BASIC STRUCTURAL UNIT OF RESNET HAVING RESIDUAL BLOCK (HE ET AL., 2016).	13
FIGURE 2.4: HISTORY OF ARCHITECTURAL INNOVATIONS OF DEEP CNNs (KHAN ET AL., 2020).	14
FIGURE 2.5: ARCHITECTURE OF LSTM (OLAH, 2015).	15
FIGURE 2.6: ENCODER-DECODER MODEL (A) WITH A TRADITIONAL APPROACH AND (B) USING ATTENTION MECHANISM (CHAUDHARI ET AL., 2019).	18
FIGURE 2.7: ATTENTION MODEL BY BAHDANAU ET AL. (2015) WHICH USES A DISTINCTIVE SEQUENCE, SINGLE-LEVEL ABSTRACTION LEVEL AND SOFT ATTENTION.	21
FIGURE 2.8: REPRESENTATION OF GLOBAL ATTENTION MODEL (LUONG ET AL., 2015).	23
FIGURE 2.9: COMPARISON OF SOFT ATTENTION (A) AND ADAPTIVE ATTENTION (B) (LU ET AL., 2017).	24
FIGURE 3.1: SAMPLE IMAGES OF FLICKR30K DATASET ALONG WITH CAPTIONS.	28
FIGURE 3.2: SKIP CONNECTION USED BY RESIDUAL NETWORKS (DWIVEDI, 2020).	29
FIGURE 3.3: RESNET152 ARCHITECTURE SHOWN USING SUMMARY () OPERATION.	30
FIGURE 3.4: ERROR RATES OF DIFFERENT STATE OF ART MODELS TRAINED ON IMAGENET DATASET (HE ET AL, 2015).	31
FIGURE 4.1: IMAGE PROCESSING USING RESNET-152 AS AN ENCODER.	38
FIGURE 4.2: ATTENTION MECHANISM TO HELP THE DECODER TO DECIDE THE NEXT WORD OF THE CAPTION.	39
FIGURE 4.3: PROCESS OF GENERATING CAPTIONS BY DECODER USING ATTENTION.	40
FIGURE 4.4: OVERALL MODEL ARCHITECTURE FOR IMAGE CAPTIONING.	42
FIGURE 4.5: WORDS GENERATED AT EACH TIMESTEP OF THE DECODER CORRESPONDING TO THE SEMANTIC MEANING OF THE IMAGE BLOCK.	46

TABLE OF TABLES

TABLE 1: PERFORMANCE EVALUATION OF VARIOUS ATTENTION MECHANISMS ON FLICKR30K DATASET WITH BEAM SIZE=3.	45
TABLE 2: PERFORMANCE EVALUATION OF VARIOUS ATTENTION MECHANISMS ON FLICKR30K DATASET WITH BEAM SIZE=5.	45
TABLE 3: CAPTIONS GENERATED BY GLOBAL AND ADAPTIVE ATTENTION MODELS.	48

1. INTRODUCTION

Artificial Intelligence has seen a huge shift in its use over the last decade for tasks such as Image Recognition, Audio Analysis, Natural Language Processing, Video Analysis, etc. This was possible due to the advancements in the semiconductor industry over the years. This made the research related to Image analysis, particularly object detection easy and attracted the attention of many researchers.

But identifying the image by detecting the objects is not enough, instead generating a description/caption which is closer to the human dialect would provide a good impression. A recent surge of developing models to generate captions for images and videos has emerged and this addresses the gap between computer vision and natural language processing.

The introduction of the Encoder-Decoder pipeline model by [Kiros et al. \(2014\)](#) was a breakthrough in image captioning. But it resulted in poor performance when the description sequence gets longer. To overcome this problem [Bahdanau et al. \(2014\)](#) proposed an “attention” mechanism that can be used as an extension to the decoder framework and used for Machine Translation tasks. Using this concept of attention [Xu et al. \(2015\)](#) were the first to propose a model to generate captions for images. And later the “attention” became an active focus of research and many researchers proposed various approaches of attention for solving tasks related to Natural language processing and Computer vision.

[Luong et al. \(2015\)](#) is one such researcher who proposed a **Global attention** mechanism, where the ground emitted words of the caption are identified to regions of an image at each time step. Similarly, an attention mechanism named **Adaptive attention** was proposed by [Lu et al. \(2017\)](#) which allows the decoder to decide whether the calculated pixels of an image need to be focused at each time step or needs to concentrate on a language model.

This research is aimed at comparing these two approaches of Global and Adaptive attention as an extension to the Encoder-Decoder framework in generating captions to the images. A deep Residual Network with 152 layers (ResNet-152) is used as an

encoder and an LSTM is used as the decoder. An evaluation of the model is performed using BLEU, METEOR, ROUGE, CIDEr metrics.

1.1 Background

Image captioning is one of the primary tasks to generate a naturalistic description of an image. This kind of task would be challenging to implement due to the involvement of major research fields like Computer vision and Natural language processing.

However this kind of tasks can play an important role in many applications like assisting visually impaired people to grasp the information around the world, scene understanding; where the Natural language and Computer vision knowledge are combined to generate Natural language descriptors based on the features available in the input image (Wang, Zhang, & Yu, 2020), in the automatic creation of metadata for images (indexing) for use by search engines, general-purpose robot vision systems (Amirian et al., 2019). This can also help in tasks related to education like machine question and answer, assisting children, etc.

The usage of the Internet in recent years has raised and it helped to connect people across the world by using many social media applications like Facebook, Instagram, Twitter, etc. and this helps them to share images/videos of the surrounding to the whole world. But the essence of images or videos shared cannot be understood by everyone who views it. This is because the image may have an object or essence specific to a category of people or location. The rapid developments in the field of AI can be used to tackle this by generating a description/caption which can be understood by different people. And active research to combine two giant fields of computer vision and Natural language processing is a long-envisioned topic.

Image captioning may sound interesting but has many challenges involved like availability of low labelled dataset, low-resolution images, issues in handling longer sentences, availability of low annotated data, etc. Because of these issues, most of the approaches are not able to generate satisfactory captions for the images. And a vast research scope is available in this area to generate an effective description of the image.

Several attempts to create an effective dataset for Image captioning tasks have been made in the past. The annotated datasets like Flickr8k, Flickr30k, MS-COCO, etc were proposed and contain rich images that can be used to train and produce a reasonably good model. But still, these datasets are limited to a few categories of objects and require high-quality annotated captions. The problems involved in handling sequences are handled by models using the Encoder-Decoder framework, but these fail to handle longer sequence captions. This is because the encoder compresses useful information of the source sequence to a fixed-length vector and the decoder finds it difficult to cope with long sentences. And to improve this, many types of research related to attention mechanisms were proposed and these can be used as an extension to the Encoder-Decoder framework.

In this experiment, the focus is provided on the Adaptive attention mechanism and Global attention mechanism, where these mechanisms are incorporated in the decoder framework to handle the longer sequence sentences.

1.2 Research problem

To perform Image captioning tasks, the Encoder-Decoder framework is preferred due to its well-suited pattern for handling sequence to sequence predictions. But the performance gets affected for longer sequences. And to improve this, **attention** mechanisms are used as an extension to the Encoder-Decoder framework. Attention mechanism generates a context vector having calculated information of pixels and using this information the decoder focuses on a particular region of an image and generates caption. [Luong et al. \(2015\)](#) proposed a state of art **Global attention mechanism** for generating the context vector where the decoder looks at the image at each timestep to generate a caption. And [Lu et al. \(2017\)](#) proposed an **Adaptive attention mechanism**, where the decoder decides whether to focus on an image or language model while generating the caption. This ultimately raises the following research question -

“To what extent the inclusion of Adaptive attention mechanism in Encoder-Decoder framework for generating the Image captions can provide the better BLEU (Bilingual Evaluation Understudy), METEOR, ROUGE, and CIDER metric score than the inclusion of Global attention mechanism?”

By observing the above Research question, we can formalise our Hypothesis as below and perform experiments:

The Generation of satisfactory captions for the images can be better achieved by including Adaptive attention mechanism as a layer between encoder and decoder when compared to the inclusion of Global attention mechanism and results in a better BLEU, Meteor, Rouge and CIDEr scores.

1.3 Research Objectives

One objective is to show that the usage of Adaptive attention mechanism instead of Global attention mechanism in Encoder-Decoder framework provides flexibility to the decoder for deciding whether the focus needs to be given to image or language model while generating the caption words.

The second objective is to perform a comprehensive literature review on various CNN architectures and attention mechanisms to understand how they can be used in the Image captioning process.

The third objective is to evaluate the Image captioning models with Adaptive attention and Global attention using BLEU, METEOR, ROUGE and CIDEr evaluation metrics.

1.4 Research Methodologies

The research methodology used is **Qualitative**. The caption generation of images is done using Adaptive and Global attention mechanisms and are compared with various evaluation metrics (BLEU, METEOR, ROUGE and CIDEr).

1.5 Scope and Limitations

The scope of this thesis is to study whether the captions generated by the model using Adaptive attention mechanism than the Global attention mechanism results in better evaluation metrics score and provides satisfactory captions.

The limitations of the study conducted are as below:

- ResNet 152 which was already trained on ImageNet dataset is used as an encoder and in this experiment, Flickr30k dataset is used, therefore encoders may not be able to detect certain categories of objects which are not present in ImageNet dataset.
- Due to constraints related to time and computation, Flickr30k dataset containing only 30k images is used for the experiment.
- Flickr30k dataset mostly depicts human activities hence the model might be slightly biased to the type of images in the dataset.
- A smaller batch size of 32 is used in the experiment. This is because fine-tuning is performed on the encoder which resulted in gradient generation and made the model large.
- Performing experiments related to high computation and hyperparameter tuning were not possible. This is because the overall model contains deep ResNet with 152 layers, attention mechanism and LSTM, which makes the model complex and requires many epochs and large numbers of images to train.

1.6 Document Outline

The rest of the document is structured as below:

Chapter 2: Literature Review

This chapter is dedicated to the literature review of previous models and approaches used for Image captioning tasks. And contains various state-of-the-art models at the respective interval of time. Initially, the chapter provides information related to various benchmark developments happening in computer vision-related tasks and in developing various CNN architectures. Later the reasons for developing attention

mechanisms, taxonomies involved are discussed along with basic structure and intuition behind its developments. Finally, the Concept of Image captioning and usage of attention mechanisms for generating captions is discussed.

Chapter 3: Design and Methodology

This chapter discusses in detail about the Business understanding and various methods used to achieve the experiment. The dataset used for the experiment and reasons for choosing this dataset is explained. Also, it discusses the reasons for choosing the particular CNN architecture, attention mechanism and decoder. And various evaluation metrics which are used for evaluating the model are discussed.

Chapter 4: Implementation and Results

This chapter discusses various data pre-processing steps followed along with the processes implemented in Encoder, attention and Decoder layers. The parameters used in these layers are discussed in detail. Followed by an overall model involving Encoder-Decoder layer along with training processes. And the results obtained during the experiment are discussed.

Chapter 5: Evaluation and Discussion

Various processes involved in evaluating and obtaining the results is discussed in this section. It also contains the important strengths and limitations identified in the obtained results.

Chapter 6: Conclusion and Future work

This section provides an overview of the experiment and the steps involved are discussed briefly. And various usages of the current experiments and the fields to which this experiment can be contributed is discussed. And finally, the future work which can be performed to improve the better usage of the research is discussed.

2. LITERATURE REVIEW

To proceed with any research, the knowledge related to previous research and understanding of various topics related to research is a must. This helps in grasping the knowledge related to the latest trends, methods, processes etc.

In this experiment, the knowledge related to various methods and approaches for achieving the task of Image captioning is obtained by conducting a thorough research review. And those can be obtained in the below sections of this chapter.

Different approaches which were used to achieve Image captioning tasks were discussed in Section 2.1 and followed by developments in CNN and LSTM are discussed respectively in Section 2.2 and 2.3.

2.1. Image Captioning

Image captioning has emerged as a recent surge and is considered as one of the prominent applications in bridging the gap between Computer vision and Natural language processing fields. And Image captioning applications play an important role in activities like assisting visually impaired people, scene understanding, having a human-robot interaction etc ([Staniūtė & Šešok, 2019](#)). Many models were proposed to achieve the task of image captioning and generally, these can be categorized into Retrieval, Template and Neural based approaches.

Initially, Image captioning was performed using Template and Retrieval based. Later due to advancements in Artificial Intelligence Neural based approaches emerged.

2.1.1 Template-based approaches

In this kind of approach, certain forced processes were used in generating the captions. Many pre-determined templates were available, and these templates contained certain fixed blank spaces, and the generated caption is filled to these spaces. Initially, the features available in the image were detected and extracted and later using processes

these detected features were filled in the available templates. [Farhadi et al., \(2010\)](#) proposed one such approach and filled the blank space within the template using the triplet of an object in the scene. [Kulkarni et al. \(2011\)](#) used a method called Conditional Random Field (CRF) which generates a graph, and nodes of this graph contain image attributes, objects and spatial relationship between them. Later these details were filled in the template available to describe the image.

Captions generated using this kind of approach were grammatically and syntactically correct sentences, this is because all the sentences were hardcoded and image features were filled in the blanks available in these sentences. The main disadvantage of this kind of approach is the captions are of **fixed length** and are **hardcoded**.

2.1.2 Retrieval based approaches

To overcome problems of the template-based approach, this approach extracts phrases from the pre-specified pool of sentences and the caption for the provided image is generated using these phrases. [Mason et al. \(2014\)](#) mapped the target image to a meaning space and later comparison is made with the sentences available in the pre-specified pool of sentences. And the semantic similarity between these sentences and image is extracted, and the sentence which is similar to the image is used as a caption for the image. [Kuznetsova et al. \(2014\)](#) proposed another approach where the images involved in the training process and the captions related to them are mapped to a common space and made them correlate. Later in other common space, the calculation is performed to identify the cosine similarity between them and the sentence with the highest score is selected as the caption for the image.

The drawback of this kind of approach is in many cases the caption retrieved may not be relevant to the image.

The Template and Retrieval based approaches are not any more effective in generating captions. Instead, the Neural based approaches are more effective and generate meaningful captions of variable length.

2.1.3 Neural based approaches

These kinds of approaches were inspired by the successful use of Encoder-Decoder frameworks in Machine translation tasks. [Kiros et al., \(2014\)](#) proposed a Feedforward approach to generate translated words. Later proposed models used a Recurrent Neural network instead of Feedforward network ([Mao et al., 2014](#); [Chen et al., 2014](#)). This framework uses deep Neural networks for generating captions and these captions are more accurate in semantic representation than compared with the approaches of Template and Retrieval. The image features are encoded into small representations using deep neural networks and passed to decoder having Recurrent Neural networks (RNN) to generate a caption for the image. Later LSTM was used in decoder instead of a vanilla RNN by [Vinyals et al. \(2015\)](#). [Karpathy & Fei-Fei \(2015\)](#) used an R-CNN as an encoder and a bidirectional RNN as a decoder to learn embedding space for a caption.

Almost every model related to Image captioning was using Encoder-decoder framework for generating captions. But for descriptions/captions with longer sequences, the performance used to be degraded. To overcome this [Bahdanau et al. \(2014\)](#) proposed an attention mechanism as an extension to the decoder framework. And later the “attention” became an active focus of research in Computer vision-related tasks, where the ground emitted words are identified to regions of an image. [Xu et al. \(2015\)](#) proposed an Encoder-Decoder model based on an attention approach for Image captioning task and this model displayed a state of the art performance. A comprehensive study between the usage of **Soft attention** and **Hard attention** based on attention access in the image was proposed. [Luong et al. \(2015\)](#) proposed similar attention named **Global attention** and **Local attention**. And many other variations of attention were proposed by various researchers.

CNN is considered to be the most preferred architecture for Encoder and LSTM is most preferred for Decoder. And the developments happening in these respective fields need to be known thoroughly for creating a better model to achieve Image captioning tasks.

2.2. Developments in CNN

To understand the content of image better, the best algorithm to look up is the CNN which have shown excellent performance in tasks related to Image segmentation, Object detection, classification and retrieval kind of tasks (Ciresan et al. 2012; Liu et al. 2019). In 1989, the work on the processing of grid-like topological data (images and time series data) by LeCuN et al. (1989) brought CNN to the forefront. Today the success of CNN's have crossed the academic scope and is an active research group in many industries and companies.

If the data which was used to train a model doesn't reflect the real world, then the algorithm/model which was trained using that data can't be considered as best. Deng et al. (2009) inspired by the above statement and using the approach of WordNet dataset started working on ImageNet project and created a huge dataset. ImageNet data consists of almost 15 million annotated images with 21 thousand groups/classes (synsets) and about 1 million images having bounding box annotations (Brownlee, 2019). In the year 2010, an annual competition using the subsets of ImageNet dataset was started called as “*ImageNet Large Scale Visual Recognition Challenge (ILSVRC)*” and was aimed at providing easy access of images to the researchers for achieving several visual recognition tasks. This competition made researchers develop benchmark CNN architectures like AlexNet, ZFNet, VGG, GoogleNet, ResNet etc. in the field of Computer vision, and is considered as one of the catalysts by many researchers for the current advancements in Artificial Intelligence field (Gershgorin, 2020; Russakovsky, 2015; Khan et al., 2020).

In the ILSVRC competition of 2012, Krizhevsky et al. (2012) enhanced the CNN architecture of LeCun et al (1995) and increased the learning potential of CNN by deepening it and introducing multiple parameter optimization strategies and called the architecture as **AlexNet** and won the 2012 challenge by achieving top 5 error rate as 16%. This is the first CNN architecture that showed pioneering results for tasks of image classification and recognition (Russakovsky et al., 2015; Khan et al., 2020). Figure 2.1 shows the basic AlexNet architecture.

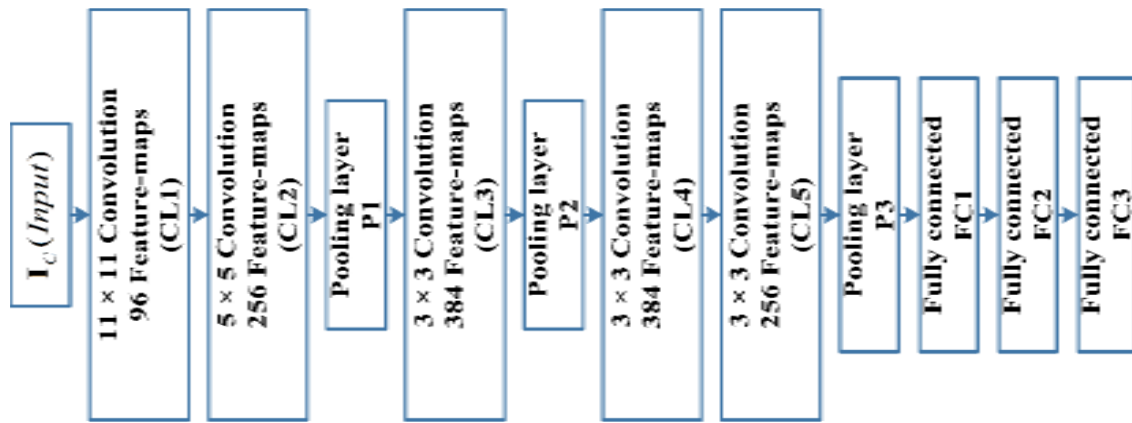


Figure 2.1: AlexNet architecture demonstrating five convolution and three fully connected layers (Krizhevsky, Sutskever & Hinton, 2012).

Until 2013, CNN's learning process was focused solely on hit and trial, without understanding the exact reason behind the improvements, this affected the performance of deep CNNs on complex images. Focusing this Zeiler and Fergus in 2013 proposed **ZfNet** (Zeiler and Fergus, 2013) also called a multilayer Deconvolutional Neural Network (DeconvNet) which won the ILSVRC-2013 competition by achieving the classification error rate of 11.2%. Zeiler and Fergus reduced stride and filter size from the initial two convolutional layers of AlexNet, which resulted in maximizing the learning of ZfNet.

In the ILSVRC-2014 competition, two benchmarks deep CNN architectures were introduced namely VGG and GoogleNet, which significantly reduced the classification error rate by almost half of the ZfNet architecture. This strengthened the confidence that image features can be learned using deep networks.

A simple and efficient design using the CNN architectures was proposed by **Simonyan and Zisserman (2015)** from the Oxford Vision Geometry Group(VGG) during the ILSVRC-2014 competition and called it as **VGGNet**. This architecture secured 2nd place in the competition and displayed the error rate as 7.3% and performed well for image classification and localization problems. Even though VGG didn't win the competition, it gained popularity for its simplicity, homogeneous topology, and increased depth. Commonly used VGG implementations for transfer learning kind of tasks are VGG16 and VGG19. The key drawback with this architecture was the usage

of 138 million parameters, which was huge and computationally expensive which made it difficult to use on low resource systems. (Khan et al., 2020)

The winner of ILSVRC-2014 competition was **GoogleNet**, also called **Inception-V1**, which was proposed by Szegedy et al. (2015). This architecture provided a top 5 classifications error rate of 6.7%. A new concept called inception block was introduced in this CNN by integrating multiscale convolutional transformations using the idea of a split, transform and merge. The architecture of this inception block is shown in Figure 2.2. A concept called auxiliary learners was also implemented in GoogleNet to speed the convergence rate. The key disadvantage of GoogleNet was its heterogeneous topology, which needs to be modified from one module to another. Another drawback with GoogleNet is it significantly reduces feature space in the next layer of the network often contributing to the loss of information. (Khan et al., 2020). Later improved versions of GoogleNet/Inception V1 were published by Szegedy et al. (2016a, 2016b) namely Inception V3, Inception V4 and Inception ResNet.

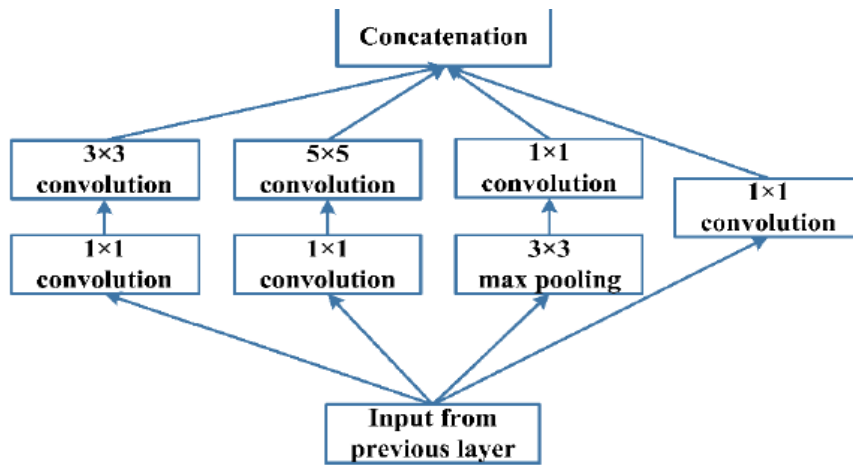


Figure 2.2: Inception block architecture used in GoogleNet showing the split, transform and merge concept (Szegedy et al., 2015).

During the ILSVRC-2015 competition, a concept of residual learning was introduced in CNNs by Microsoft Research team, which revolutionized the architectural race of CNN and named it as ResNet (He et al., 2016). This architecture dragged the top 5 error rate to 3.6% and won the competition. It contained 152 layers which were stacked using residual blocks.

ResNet (Residual Networks)

In recent years ResNet or Residual networks are considered to be the most ingenious architecture in the field of computer vision. It was proposed by He et al. (2016) and contained 152 layers of deep CNN and is the winner of the ILSVRC 2015 competition by displaying the error rate of 3.6%.

Before ResNet, the training of deep networks was hard due to the vanishing gradient problem. The introduction of “identity shortcut connection”, also called residual blocks made the network to handle this problem. These blocks are the combination of conv-relu-conv layers and the network follow skip connections between these blocks. The basic unit of these residual blocks is shown in Figure 2.3.

ResNet proposed by He et al. (2016) was 20 times deeper than the AlexNet and 8 times than that of

VGGNet. And the complexity related to computation was less when compared with these previous models (Krizhevsky et al., 2012; Simonyan and Zisserman, 2015). Also, He et al. experimentally showed that when compared with the 34-layer plain Net for the Image classification task, Resnet with 50/101/152 layers were displaying less error rate.

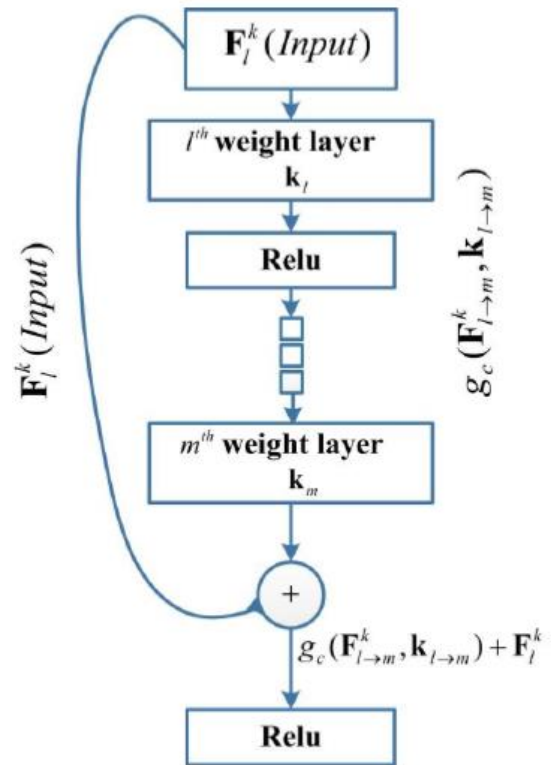


Figure 2.3: Basic structural unit of ResNet having Residual block (He et al., 2016).

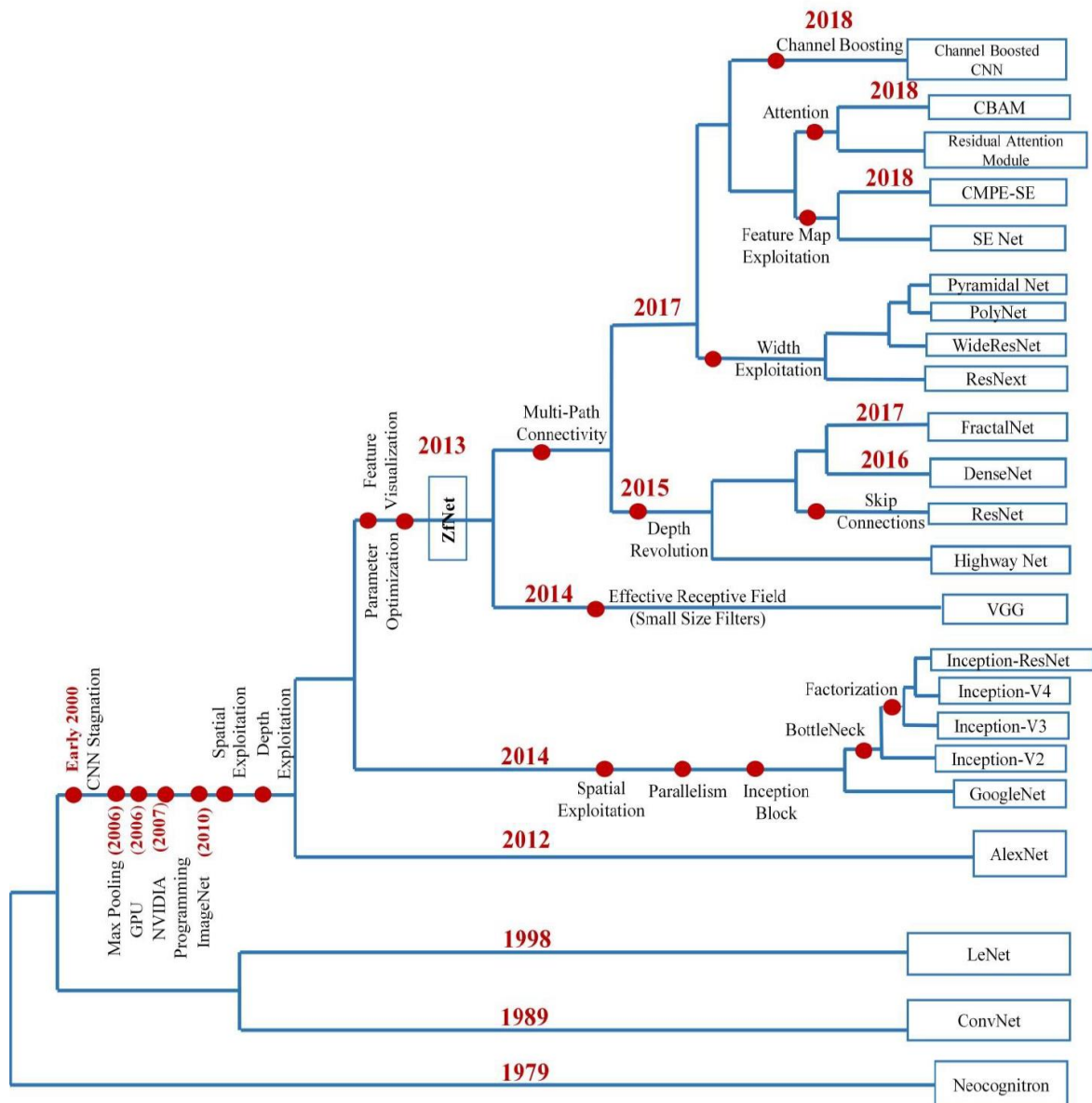


Figure 2.4: History of architectural innovations of deep CNNs (Khan et al., 2020).

2.3. Developments in LSTM

The tasks which required handling of sequential data used to adopt Recurrent Neural Networks (RNN). But this kind of approach was fine with tasks having short term dependencies but when dealing with tasks having long term dependencies, the RNNs don't work well. This is because the gradients of subsequent nodes decrease with increase in time step and this leads to difficulty in providing updates to the preceding nodes (Hochreiter et al., 2001). And dealing with tasks having long term dependencies

using RNN is a problem and the accuracy of the predictions would be poor. To overcome these challenges, [Schmidhuber and Hochreiter \(1997\)](#) proposed the LSTM model, which have gate functions introduced to cell structure. This made LSTM tackle the problems faced by RNN and handle long term dependencies.

These LSTMs consist of internal mechanisms called gates, which help to regulate the information flow. Also consists of different memory blocks called ‘cells’, which are used for remembering things and manipulations to these memories are performed using different ‘gates’ of LSTM.

The control flow in LSTM is similar to that of RNNs except with the gates. The basic architecture of LSTM is shown in Figure 2.5.

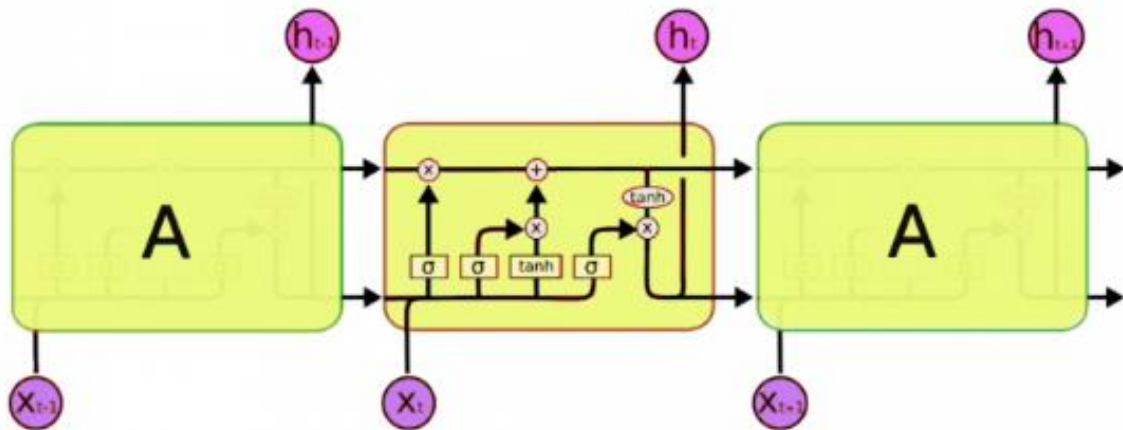
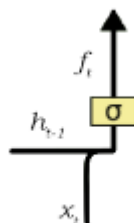


Figure 2.5: Architecture of LSTM (Olah, 2015).

The different gates available in LSTM are Forget gate, Input gate and Output gate.

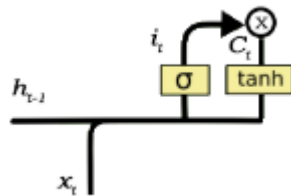
Forget gate:



This gate decides whether the available cell information needs to be removed or not. The information from the previous hidden state (h_{t-1}) and current input (x_t) is taken as input to Forget gate. These obtained inputs are multiplied with weight matrices and a

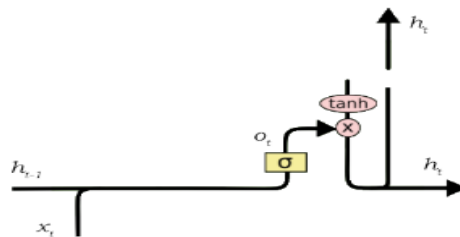
bias is added and passed through a sigmoid function, which outputs either 0 or 1. If the output obtained is 0 then the information in the cell is removed and if 1 then the information is restored and the output from the sigmoid function is multiplied to cell state.

Input gate:



This gate is responsible for adding information to the cell state. It consists of 2 functions namely sigmoid and tanh as shown in the above image. The previous hidden state and current input are passed as inputs to these functions. The sigmoid function outputs a value ranging from 0 and 1. Where 0 means important and 1 is not important. Whereas tanh function squeezes values between -1 and 1, this helps in the regulation of the network. Later both the outputs from tanh and sigmoid functions are multiplied, where the sigmoid output decides which information of tanh is important to be added to the cell state.

Output gate:



This gate helps in deciding the next hidden state. A previous hidden state and current output are passed to the sigmoid function of this gate. Parallely a cell state which was recently modified is passed to a tanh function. Now the output of sigmoid and tanh functions are multiplied, this decides what information needs to be carried by the next hidden state.

Due to these advantages of LSTM, many researchers were inspired to use LSTM in their models to solve tasks related to long term dependencies. Later to achieve

sequence to sequence tasks [Kiros et al. \(2014\)](#) proposed an Encoder-Decoder framework, which became a breakthrough in handling tasks like Machine Translation, Question and Answer, Chatbots, Image captioning etc. But as the sequences got longer the performance of the models got decreased.

Later the introduction of attention mechanism by [Bahdanau et al. \(2014\)](#) in Machine translation task as an extension to Encoder-Decoder framework handled this problem of longer sequences by focusing on certain important words in source sentence to generate sentences in the decoder. This became active research and many types of research were proposed using an attention mechanism.

2.3.1 Attention mechanism

The basic human understanding of the word attention means selectively focusing on an object/thing by neglecting/overlooking the other. With the same idea, [Bahdanau et al. \(2014\)](#) proposed an approach using attention and used it in Machine translation tasks. Earlier a fixed-length vector was used in an encoder-decoder network for generating the translated sentence, but it was a bottleneck and was affecting the performance of the encoder-decoder network. In this approach researchers allowed the model to automatically identify certain words in source sentence which can be useful to generate a translated sentence. Within AI community, attention approach became incredibly common and soon it was used in many applications related to Natural language processing ([Galassi et al., 2020](#)), Speech ([Cho et al., 2015](#)) and computer vision ([Xu et al., 2015](#); [Wang and Tax, 2016](#)).

In certain tasks, few aspects of input would be more important than others. For example, in tasks related to sentiment analysis and translation, few words in the input are important to predict the next word ([Yang et al., 2016](#)). Also in tasks related to Image captioning importance needs to be given to a particular section of the input image so that it would be helpful to generate a meaningful caption for the image ([Xu et al., 2015](#)). This principle of relevance is integrated by Attention mechanism and help the models to pay attention selectively to certain parts of input and complete the task effectively ([Chaudhari et al., 2019](#)).

Attention mechanism got introduced to mitigate the challenges posed by a traditional encoder-decoder network like loss of information due to a single fixed-length vector and aligning model between input and output sequences (Cho et al., 2015). It overcomes these issues by inducing attention weights over input sequence and predicting the next output token. The architecture of traditional encoder-decoder model and the model using attention is shown in figure 2.6.

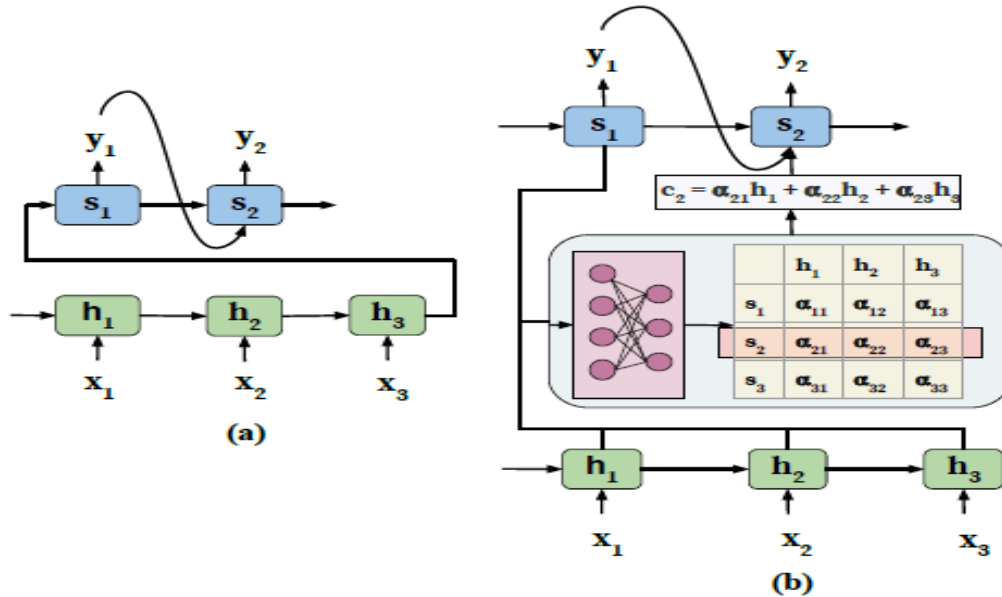


Figure 2.6: Encoder-Decoder model (a) with a traditional approach and (b) using attention mechanism (Chaudhari et al., 2019).

The attention block in Figure 2.6 (b) learns the attention weights ‘ α_{ij} ’ using the information related to the encoder hidden state ‘ h ’ and decoder hidden state ‘ s ’.

Mainly there exist three salient neural architectures which use attention mechanism:

(1) **Encoder-Decoder Framework** – Attention mechanism mostly prefers and uses this kind of architecture. Mostly CNN is used as encoder and RNN or LSTM is used as a decoder in this framework. Any input representation can be used, and they can be reduced to a single fixed-length context vector by attention layer and pass to the decoding step. (Bahdanau et al., 2014; Vinyals et al., 2015; Yang et al., 2016).

(2) **Transformer** – The sequential processing of input followed in the encoder-decoder network is computationally expensive and to address this issue Vaswani et al. (2017) proposed transformer model which uses self-attention to detect the dependencies

between the input and output. Also, this architecture eliminates sequential processing and promotes parallel processing.

(3) **Memory Networks** – Certain applications like Question Answering and chatbots learn by utilizing the information which is available in the databases. The database and query form the input to the network and certain facts would be closely associated with the query when compared with others. To achieve this kind of approach [Sukhbaatar et al. \(2015\)](#) proposed End-to-End Memory Networks which stores facts using an array of memory blocks and to get the relevance of each fact attention is used.

Taxonomy in attention:

Attention is considered in four broad categories based on the application of interest.

The first category is based on the **number of sequences**: where different types of attention like distinct, co-attention and self – attention exists. Applications, where a single input sequence is used to generate respective output sequence, can usually be referred to as **distinctive** where encoder hidden state and decoder hidden state belongs to distinct input sequence and output sequence. Usage of this kind of attention approach can be seen in applications like Machine translation ([Bahdanau et al., 2014](#); [Luong et al., 2015](#)), Speech recognition ([Chan et al., 2016](#); [Cho et al., 2015](#)) and Image captioning ([Xu et al., 2015](#)). Applications whose models learn the attention weights with multiple input sequences jointly and captures the interactions of these inputs can be considered as **co-attention**. This type of attention is used in applications related to Visual Question answering ([Lu et al., 2016](#)). And **self-attention** can be used for tasks where input is a sequence, but the respective output is not a sequence, thus allowing the model to enable each word in the input to be associated with other words in the input. This kind of attention can be observed in tasks related to text classification and recommendation ([Wang et al., 2016](#)).

The second category is based on the **number of abstraction levels**: Here the attention mechanism can be of type **Single-level attention** and **Multi-level attention**. In Single level attention type, the attention weights are calculated using the original input sequence ([Bahdanau et al., 2014](#); [Xu et al., 2015](#)) whereas attention extended sequentially to several levels of abstractions (dimensions) of input sequence can be

called as Multi-level attention (Lu et al., 2016; Yang et al., 2016). Further based on how the learning of weights occurred, the multi-level attention is classified into ‘top-down’ (Aneja et al., 2018; Hendricks et al., 2016; Zhao and Zhang, 2018) and ‘bottom-up’ (Yang et al., 2016).

Yang et al. (2016) proposed a model which uses attention mechanism (Multi-level) at two levels namely at word level and sentence level and called the model as “Hierarchical Attention Model” (HAM). This model grasps the documents hierarchical structure (Document is composed of sentences, and sentences are further composed of words) and derives/extracts words which play important role in sentences and then the sentences which are important in a document.

The third category is based on the **number of positions**: Where the variations emerge in determining attention function based on the position/location of the input sequence. There exist many attentions under this category like soft, hard, global and local. The attention mechanism which was used by Bahdanau et al. 2015 is **soft attention**, where to build a context vector, a weighted average of features across all hidden states of input sequence was used. And Xu et al. (2015) proposed an attention mechanism which was used for Image captioning task called as **Hard Attention**, where the computation of context vector is done based on stochastically sampled hidden states of input. This approach required less calculation than soft attention at inference time. The **Global** and **local attention** mechanisms were proposed by Luong et al. (2015) for task related to machine translation, where the global attention was similar to soft attention but the local attention was a mixture of both hard and soft attentions. In local attention, the model first detects a single aligned location from the input sequence for the current target word and a window around that location is picked to create local attention. Figure 2.6 shows the architecture of attention block used by Bahdanau et al. 2015 where attention having a distinctive sequence, single-level abstraction and soft attention is used.

The fourth category is based on the **number of representations**: In this category attention mechanisms like Multi representational Attention and Multi-dimensional attention can be considered. In most applications usually, the representation of a single feature of the input sequence is used. But for certain downstream tasks, the usage of

single feature representation may not be enough and instead, we may require assigning attention weights accordingly (Chaudhari et al., 2019). Different features of the input sequence are captured and using multiple feature representations the attention weights are assigned to these different representations, such kind of model can be called as **Multi representational attention** model. Kiela et al. (2018) proposed multi representational model to improve sentence representations where the attention weights were trained over different word embeddings of the same input sentence. Similarly, Maharjan et al. (2018) proposed a model where attention was used to represent different feature representations of books by dynamically weighing these representations and capturing information like lexical, syntactic, visual and genre. In the case of **Multi-dimensional attention**, the intuition is the same as multi representational attention, but the weights are induced to assess the importance of each dimension of the input vector. This kind of approach is extremely effective for applications related to natural language processing to handle the polysemy problem (Chaudhari et al., 2019). Lin et al. (2017) used this approach to generate effective sentence embedding representation and Shen et al. (2018) used this approach in their model for language understanding problem.

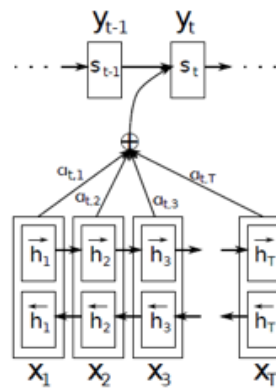


Figure 2.7: Attention model by Bahdanau et al. (2015) which uses a distinctive sequence, single-level abstraction level and soft attention.

2.3.2 Usage of attention mechanism as an extension to Decoder for achieving Image Captioning task

The introduction of the encoder-decoder pipeline model by Kiros et al. (2014) was a breakthrough in image captioning. And with the usage of attention mechanism by

[Bahdanau et al. \(2014\)](#) in Machine translation task, the “attention” became an active focus of research in Computer vision-related tasks, where the ground emitted words are identified to regions of an image. [Xu et al. \(2015\)](#) proposed an encoder-decoder model based on attention approach for Image captioning task and this model displayed a state-of-the-art performance. A comprehensive study between the usage of **Soft attention** and **Hard attention** based on attention access in the image was proposed. Initially, the image was encoded by a CNN and the image features were extracted and passed to a decoder with LSTM to generate the caption whereas an attention mechanism was used to learn the weights for this task.

Using Soft and Hard attention [Luong et al. \(2015\)](#) proposed similar attention named **Global attention** and **Local attention**, where Global attention was similar to Soft attention but with differentiation in considering the hidden states of encoder and decoder, whereas Local attention was a mixture of Soft and Hard attention. In the case of the model with local attention, the single aligned position of the target word is predicted and at source position, a window is centred to compute the context vector.

A review network was used by [Yang et al. \(2016\)](#) to retrieve global properties present in a compact vector representation and are passed to a decoder having attention mechanism. [Yao et al. \(2015\)](#) proposed a temporal attention mechanism to focus keyframes of the video which can be considered relevant with a predicted word. A semantic concept was extracted using K-NN and multi-label ranking and later combined them with RNNs to generate captions by fusing them to a vector using attention mechanism ([You et al., 2016](#)). Not all the words of the caption can be tied to the regions of the images instead a model determines when to depend on the image region and when to depend on the language model. To achieve this [Lu et al. \(2017\)](#) proposed an adaptive attention model to generate captions for the image.

2.3.2.1 Global Attention mechanism for Image captioning

Earlier attention mechanisms before Global attention were using the output of the decoder from the prior time step to decide the important sequence (context vector) in the encoder. Whereas the Global attention considers the output of encoder and decoder for the current time step for deciding the important sequence (context vector), it also considers all of the encoder’s hidden states. Even though initially Global attention was

developed for machine translation tasks, they can be used in the Image captioning task as well.

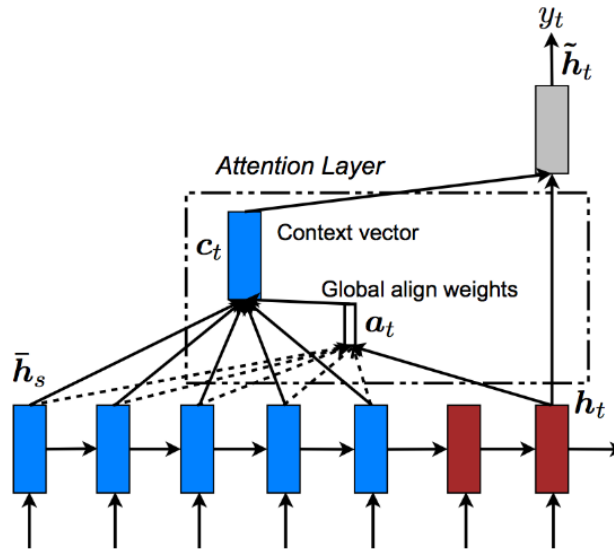


Figure 2.8: Representation of Global attention model (Luong et al., 2015).

To derive Context vector (C_t), consideration of all the encoder hidden states is done by Global attention. This can be observed in Figure 2.8 shown above. A variable-length Global alignment vector (a_t) is computed using similarity measure between source hidden state (h_t) and target hidden state (\bar{h}_s). The computation of Global alignment vector helps in deriving context vector.

A score is calculated at each encoder time step using target decoding. This helps to find similarities between target hidden state and source hidden state. The scoring functions like dot, general, concat as shown below can be used.

$$\text{score}(h_t, \bar{h}_s) = \begin{cases} h_t^\top \bar{h}_s & \text{dot} \\ h_t^\top W_a \bar{h}_s & \text{general} \\ v_a^\top \tanh(W_a [h_t; \bar{h}_s]) & \text{concat} \end{cases}$$

In this part of the experiment, a simple dot function is used to compute the score. Once the scores are generated then they are normalized using a SoftMax function. After which a Global Alignment vector is computed by calculating the weighted sum and finally a context vector is generated.

The decoder uses this context vector along with target decoding and later using tanh function these are weighed and transferred. And this final decoder is passed to a SoftMax function, which helps to predict the next word probability.

2.3.2.2 Adaptive Attention mechanism for Image Captioning

While generating each word of the caption, most of the attention models try to look at the pixels of image actively at each time step of the decoder. However, while generating non-visual words like ‘and’, ‘the’, ‘of’ etc, the decoder doesn’t require visual attention of image to generate a word instead it can rely on a language model. This can be achieved by using Adaptive attention. This also helps the model from misleading and reducing the effectiveness of these signals from the image.

The Adaptive attention can be considered as an extension of spatial attention which knows when to depend on visual signal and when not. The spatial attention consists of an updated LSTM, which produces a visual sentinel vector instead of a single hidden state. This helps the model to increase the latent representation of decoder memory. Adaptive attention makes use of this spatial attention and extracts a new component called “Visual sentinel”. This provides the model with a gate, which helps it to decide whether to attend an image or not.

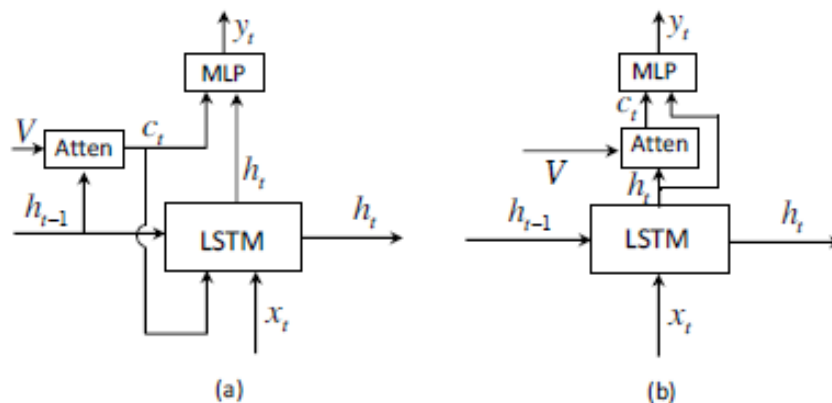


Figure 2.9: Comparison of Soft attention (a) and Adaptive attention (b) (Lu et al., 2017).

In case of Adaptive attention, a current hidden state h_t is used to analyse where to look in the image, whereas in previous attention mechanisms like soft attention it was

absent. This can be observed in Figure 2.9. Thus, a context vector is generated and by combining both information sources, the next likely word is generated.

For sentinel gate generated in Adaptive attention can be obtained using the below formula. S_t is considered as visual sentinel.

$$g_t = \sigma (W_x x_t + W_h h_{t-1})$$

$$s_t = g_t \odot \tanh(c_t)$$

And to obtain a context vector sentinel visual information and spatial information are combined as shown in below formula.

$$\hat{c}_t = \beta_t s_t + (1 - \beta_t) c_t$$

Where c_t is a context vector and β_t is sentinel gate.

And finally, the probability of word at time t can be calculated using

$$p_t = \text{softmax} (W_p (\hat{c}_t + h_t))$$

Where W_p is considered as Weight parameters which are to be learnt. These processes help the model to adaptively look at image whenever required and generates the next word of the caption.

2.4. Summary, Limitations and Gaps in the Literature

A detailed review of various state of art approaches related to Image captioning, CNN architectures, attention mechanism and LSTM networks was carried out.

The main challenge observed is the unavailability of a vast dataset with rich annotated texts. Only a few datasets related to Image captioning is available and these datasets are biased towards particular categories of data.

The literature review also explored previous approaches which were followed earlier to achieve Image captioning along with the latest approaches. It also provided the various developments happened in the development of current CNN architectures and LSTM architectures. It is also observed that many advancements in the attention mechanism occurred in the field of Natural language processing and these can be utilized in Computer vision field as well. No many papers related to a comparative study of various attention mechanisms in the field of Image captioning is observed.

3. DESIGN AND METHODOLOGY

This section describes the experiments carried out to determine whether the use of Adaptive attention mechanism in an encoder-decoder framework is better than the use of Global Attention mechanism. Usage of attention mechanism for Image captioning tasks using Encoder-Decoder framework is widespread and this helps models to choose the certain location of encoding which it thinks as relevant for generating words/captions.

In the case of Global attention, the model mostly attends to the image at every time step, regardless of which word will be generated next. Whereas in adaptive attention, the model decides when to rely on visual signals and when on a language model. Images which depicts various activities performed by humans and animals is considered for this experiment. Most common datasets used for Image captioning activities are MSCOCO, Flickr30k and Flickr8k. The dataset which is used in this experiment, the data preparation and processing steps are described in the below sections.

The Encoder-Decoder model architecture along with the Global and Adaptive attention mechanisms followed by Evaluation metrics are also presented. Explanation of convolutional network and ResNet architecture used for encoding image features and the usage of LSTM as a decoder to decode the encoded images is discussed in detail. Performance evaluation of models over validation datasets is well documented.

Due to Resource constraints, all the experiments were performed on Kaggle cloud service having 16GB Nvidia P100 GPU with python version 3.7.6 and PyTorch 1.7.0

3.1 Business Understanding

The objective of this research is to show that the models which use the attention mechanism can capture the essence of the entire image and generates a satisfactory caption. And the usage of Global attention provides importance to all the inputs and considers the hidden states of ResNet encoder and LSTM decoder while generating

captions and attends the image signals at every time step. Whereas the inclusion of Adaptive attention over Global attention provides the model with the flexibility to automatically decide when to rely on image signals and when on a language model.

3.2 Hypothesis

H₁: The Generation of satisfactory captions for the images can be better achieved by including Adaptive attention mechanism as a layer between encoder and decoder when compared to the inclusion of Global attention mechanism and results in a better BLEU, Meteor, Rouge and CiDER scores.

H₀: The Generation of satisfactory captions for the images cannot be better achieved by including Adaptive attention mechanism as a layer between encoder and decoder when compared to the inclusion of Global attention mechanism and results in a worse BLEU, Meteor, Rouge and CiDER scores.

3.3 Understanding of Data

There exist certain benchmark datasets like MSCOCO, Flickr30k, Flickr8k etc having varying image count for the task of image captioning. In this experiment due to constraints of time and computational power, **Flickr30k** dataset has been used.

Flickr30k dataset:

This is one of the most commonly used datasets for Image captioning task which is rich in the number of images and contains 31,783 images which are collected from Flickr website. Mostly these images depict human activities on various tasks. Each image is provided with five crowdsourced captions. And a total of 158,915 captions are available for 31,783 images (Young et al., 2014). Later using this dataset 244k coreference chains along with 276k manually annotated bounding boxes were developed (Plummer et al., 2015). As this experiment requires only images and their respective captions, the coreference chains and annotated boxes are neglected. Sample images of Flickr30k along with their respective crowdsourced captions can be seen in Figure 3.1.



Two friends enjoy time spent together .
A man in a blue shirt standing in a garden .
Two men in green shirts are standing in a yard .
Two young , White males are outside near many bushes .
Two young guys with shaggy hair look at their hands while hanging out in the yard



Three men on a large rig .
Four men on top of a tall structure .
Two men working on a machine wearing hard hats .
Workers look down from up above on a piece of equipment .
Several men in hard hats are operating a giant pulley system .



A girl going into a wooden building .
A little girl climbing into a wooden playhouse
A little girl climbing the stairs to her playhouse .
A little girl in a pink dress going into a wooden cabin .
A child in a pink dress is climbing up a set of stairs in an entry way .

Figure 3.1: Sample images of Flickr30k dataset along with captions.

3.4 ResNet-152 as Encoder

Generally, a model that produces sequences is used as an encoder to encode the input. Encoder steps through the input sequence at time steps and encodes them into a fixed-length vector called a Context vector. In this experiment the images are used as input, therefore the best model to encode the images is **Convolutional Neural Networks (CNNs)**. The encoder output obtained provides a summary representation of information that is considered to be useful in the image. To understand the essence of the image very well, the model needs to be trained effectively on various categories of images and should be exceptionally efficient in classifying the images. And building such a model from scratch would be expensive and requires more computation. For years, many people are building extraordinary models for tasks like image classification, and borrowing such already trained models and fine-tuning them to achieve our task would be helpful. This kind of approach is called **Transfer learning**. And in this experiment, a 152 layered Residual Network (ResNet 152) that was well-trained on the ImageNet classification task is used as an encoder.

Residual Networks (ResNet 152):

Residual Networks are classic Neural networks, which are considered as a pillar for many tasks related to computer vision. During the ILSVRC 2015 challenge, the Residual Networks outperformed other participants and was declared as the winner (He et al., 2015). Earlier to ResNets the deep neural network models were facing the difficult problem of vanishing gradients and the accuracy was getting saturated and degrading rapidly. But ResNets allows training deep neural networks successfully with 150+ layers by using the concept of Residual blocks (skip connection).



Figure 3.2: Skip connection used by Residual Networks (Dwivedi, 2020).

We can see in figure 3.2 that the convolutional layers are stacked together for both the images. In image having skip connection, we can observe the addition of original input to the convolutional layers output block. This kind of arrangement helps gradient to flow along the alternate shortcut path and mitigates the problem of vanishing gradient. ResNet uses batch normalization, this helps the input layer to get adjusted and increase the performance of the network.

ResNet152 is a combination of 151 convolutional blocks and one Fully connected layer stacked over one another using skip connections as discussed in the above paragraph. Each ResNet block of ResNet 152 is 3-layers deep. Many of such 3 layers deep ResNet blocks are constructed. The basic architecture of ResNet 152 is shown in Figure 3.3. We can observe that smaller and smaller representations of the original image are progressively created and using a greater number of channels the subsequent representations are learned well. Encoding of size 14*14 with 2048 channels is produced by ResNet 152.

```

Model: "resnet152"
-----
Layer (type)                Output Shape                Param #    Connected to
-----
input_1 (InputLayer)        [(None, None, None, 0
conv1_pad (ZeroPadding2D)   (None, None, None, 3 0    input_1[0][0]
conv1_conv (Conv2D)         (None, None, None, 6 9472 conv1_pad[0][0]
conv1_bn (BatchNormalization) (None, None, None, 6 256 conv1_conv[0][0]
conv1_relu (Activation)     (None, None, None, 6 0   conv1_bn[0][0]
pool1_pad (ZeroPadding2D)   (None, None, None, 6 0   conv1_relu[0][0]
pool1_pool (MaxPooling2D)   (None, None, None, 6 0   pool1_pad[0][0]
conv2_block1_1_conv (Conv2D) (None, None, None, 6 4160 pool1_pool[0][0]
conv2_block1_1_bn (BatchNormali (None, None, None, 6 256 conv2_block1_1_conv[0][0]
conv2_block1_1_relu (Activation (None, None, None, 6 0   conv2_block1_1_bn[0][0]
conv2_block1_2_conv (Conv2D) (None, None, None, 6 36928 conv2_block1_1_relu[0][0]
conv2_block1_2_bn (BatchNormali (None, None, None, 6 256 conv2_block1_2_conv[0][0]
-----
|
|
|
-----
conv5_block2_out (Activation) (None, None, None, 2 0   conv5_block2_add[0][0]
conv5_block3_1_conv (Conv2D) (None, None, None, 5 1049088 conv5_block2_out[0][0]
conv5_block3_1_bn (BatchNormali (None, None, None, 5 2048 conv5_block3_1_conv[0][0]
conv5_block3_1_relu (Activation (None, None, None, 5 0   conv5_block3_1_bn[0][0]
conv5_block3_2_conv (Conv2D) (None, None, None, 5 2359808 conv5_block3_1_relu[0][0]
conv5_block3_2_bn (BatchNormali (None, None, None, 5 2048 conv5_block3_2_conv[0][0]
conv5_block3_2_relu (Activation (None, None, None, 5 0   conv5_block3_2_bn[0][0]
conv5_block3_3_conv (Conv2D) (None, None, None, 2 1050624 conv5_block3_2_relu[0][0]
conv5_block3_3_bn (BatchNormali (None, None, None, 2 8192 conv5_block3_3_conv[0][0]
conv5_block3_add (Add) (None, None, None, 2 0   conv5_block2_out[0][0]
conv5_block3_out (Activation) (None, None, None, 2 0   conv5_block3_add[0][0]
avg_pool (GlobalAveragePooling2 (None, 2048) 0 conv5_block3_out[0][0]
-----
Total params: 58,370,944
Trainable params: 58,219,520
Non-trainable params: 151,424

```

Figure 3.3: ResNet152 architecture shown using summary () operation.

Even after the increase in the depth of layers in ResNet 152, the complexity is less when compared with VGG models and an error rate produced by ResNet 152 is very less compared to other benchmark models like AlexNet, VGG 16/19, GoogleNet. Figure 3.4 shows that the top 5 error rate of ResNet152 model is 3.57%, which is very little compared to other models.

Model	Size (M)	Top-1/top-5 error (%)	# layers	Model description
AlexNet	238	41.00/18.00	8	5 conv + 3 fc layers
VGG-16	540	28.07/9.33	16	13 conv + 3 fc layers
VGG-19	560	27.30/9.00	19	16 conv + 3 fc layers
GoogleNet	40	29.81/10.04	22	21 conv + 1 fc layers
ResNet-50	100	22.85/6.71	50	49 conv + 1 fc layers
ResNet-152	235	21.43/3.57	152	151 conv + 1 fc layers

Figure 3.4: Error rates of different state of art models trained on ImageNet dataset (He et al, 2015).

Due to less error rate and complexity involved when compared with other state of art models and also considering the vanishing gradient problem, ResNet with 152 layers has been considered for this experiment and fine-tuning is done to match the image captioning task.

3.5 Attention mechanism for Image Captioning

The Encoder-Decoder framework offers a pattern for solving difficult problems with sequence to sequence predictions, but the performance gets affected for longer sequences. This is because the encoder compresses useful information of source sequence to a fixed-length vector and decoder finds it difficult to cope with long sentences. And to improve this, **attention** mechanisms can be used as an extension to Encoder-Decoder framework. Predominantly attention is used in Natural language processing tasks like Machine Translation, Question answering, chatbots etc. In tasks like Machine translation, certain words are considered as important than others, whereas in Image captioning tasks certain pixels of the image are considered important

than others. And using attention mechanism in Image captioning task helps the model to identify the parts/pixels of the image from encoder which are relevant and passes them to Decoder to generate a meaningful caption for the image.

And the understanding of sequences produced so far by the encoder plays an important role in determining which portion of the pixels is important in generating captions. The attention mechanisms thus understand the sequence generated from an encoder and attend to the image at each time step to decide the word to be generated next. There exist many attention mechanisms and each mechanism uses a different way of deciding the important part of the sequence. As part of this experiment, the captions for the image is generated using the Global attention mechanism and Adaptive attention mechanism.

3.6 LSTM as Decoder

The responsibility of the decoder is to read the context vector generated by the encoder and steps through output time. In this experiment, the decoder looks at different parts of the image through the attention mechanism and predicts the words for the caption. In the case of sequence prediction tasks, the LSTMs (Long Short-Term Memory) are capable of learning order dependency for large time steps and therefore LSTM is used as a decoder in our experiment.

Long Short-Term Memory (LSTM)

LSTMs are considered as a kind of RNN which can acquire context information effectively from longer sequences. Whereas Recurrent Neural Networks (RNN) works fine with tasks having short term dependencies but when dealing with tasks having long term dependencies, the RNNs doesn't work well. This is because the gradients of subsequent nodes decrease with increase in time step and this leads to difficulty in providing updates to the preceding nodes. And dealing with tasks having long term dependencies using RNN is a problem and the accuracy of the predictions would be poor. To overcome these challenges, LSTMs can be used.

In this experiment, LSTMs are used as a language model basis and respective attention mechanisms are connected to the LSTM to extract image features/pixels required for

caption generation. A SoftMax layer of an encoder is removed and vector from the fully connected layer is used in selectively focusing pixels/locations of the image by embedding the attention layer.

3.7 Evaluation metrics

BLEU, METEOR, ROUGE, CIDEr and SPICE are considered to be the general evaluation metrics for the tasks related to sentence generation. These metrics after generating the sentences calculates the consistency of n-gram between them. In this experiment **BLEU, METEOR, ROUGE** and **CIDEr** metrics are considered in evaluating the generated predicted captions.

BLEU (Bilingual Evaluation Understudy)

BLEU is a metric which was originally intended for Machine Translation kind of tasks. The n-gram between the predicted statement and reference statement is obtained and the correlation between them is analysed. In quality-wise, it is considered to be a highly correlated with human judgements, where the translated statement is compared with a professional human translation statement and if it finds closer, then the performance can be considered as better. In this experiment, the processing would be the same as Machine translation, where images can be equivalent to multiple source sentences. The advantage of BLEU is that it uses n-gram as granularity instead of a word. BLEU provides a score in the range of 0 and 1, where the score nearer to 1 means more common with human reference translations.

In this experiment, BLEU is used for 2 tasks, one to stop the training process when the BLEU score gets degraded and second for evaluating the performance of the model on validation data.

METEOR (Metric for Evaluation of Translation with Explicit Ordering)

This metric was originally intended for Machine Translation tasks. The reference translation along with model generated translation is aligned and matched with accuracy, recall and F-value. Along with word matching, it consists of additional features like stemming and synonym matching, which are not found in other metrics. This metric is a harmonic mean of recall and unigram precision which was designed to overcome the problems of BLEU metric like producing a good correlation at sentence

or segment level for human judgement. As the METEOR score gets higher, the performance of the model would be better.

ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

It was intended for evaluating tasks like Machine translations and automatic summarization of texts. Reference sentences are compared with the translations obtained from the model. It consists of four different measures namely ROUGE_L, ROUGE-N, ROUGE-S, and ROUGE-W. A statistics of the Longest Common Subsequence (LCS) is considered by ROUGE_L, where a sentence level similarity is taken into consideration and n-grams with longest co-occurring are identified. N-gram in ROUGE_N is calculated using a recall obtained from reference summaries and candidate summary. Skip- bigram overlapping of model generated sentence and reference summaries is measured in ROUGE_S. As the ROUGE score increases the performance of the model also increases.

CIDEr (Consensus-based Image Description Evaluation)

It was intended specifically for tasks related to image annotation. In this metric for each n-gram, a Term Frequency-Inverse Document Frequency (TF-IDF) weight calculation is calculated, this helps in measuring the consistency of annotated image. A TF-IDF vector is used for representing each sentence and a score is generated by calculating cosine resemblances of reference data and sentence generated by the model. As the CIDEr score increases, the performance of the model increases.

4. IMPLEMENTATION AND RESULTS

The implementation details along with processes to train the model are provided in this section. Flickr30k dataset has been used for performing the experiment and other methods used for implementation purpose are discussed in section 3. An Encoder-Decoder Framework has been used for this experiment. The implementation of the Encoder model, Attention models and Decoder with attention is discussed separately in the below sections and an overall model architecture followed by results is provided at the end of this section.

4.1 Data pre-processing

Training and Validation split:

A split which was proposed and used by [Karpathy & Fei-Fei \(2015\)](#) was used to split the training, validate and test data. A total of 28k images for training and 1000 images for validation and test process respectively was used in this experiment.

Images:

This experiment uses a pre-trained model (ResNet 152) as Encoder and the images which are available from the Flickr30k dataset can be of variable dimension. Therefore, we need to process the images to the dimensions which are compatible with the pre-trained encoder model.

To achieve this,

- The pixel values of the images were transformed to the range [0,1]
- The image was normalized using mean and standard deviation of ImageNet images RGB channels. (mean = [0.485, 0.456, 0.406] and std = [0.229, 0.224, 0.225])
- To maintain uniformity, all the Flickr30k images are resized to 256*256 dimension.
- As PyTorch follows NCHW convention, the images are converted to a Float tensor of dimension 'B, 3, 256, 256', where B refers to batch size.

Captions:

An LSTM network is used to generate caption in the decoder and to generate a word, the decoder requires another word. Therefore, the Image captions available in Flickr30k dataset acts as both input and target to the model's decoder framework.

- We require a starting word and ending word in the caption so that the decoder learns to predict the start and end of the caption. To achieve this, the word “<start>” is appended at the zeroth position and word “<end>” is appended at last position of the caption. This also helps the model to understand when to stop decoding.
- As fixed size tensors are used to pass the captions, and each caption will be of varying length, the captions are padded by using the word “<pad>”, such that each caption would be of the same length.
- Words which occur less than 5 times in the whole corpus are replaced with ‘<unknown>’
- Further an index mapping of each unique word from the captions is created (word_map), this includes the words <start>, <end>, <unknown> and <pad>.
- Captions which are passed to the model as input is maintained as an Integer Tensor of padded length.
- Captions whose length is longer than 100 words are not sampled.

After the pre-processing of Flickr30k captions with the above steps, a vocabulary size of 23,457 words has been observed.

4.2 ResNet 152 as Encoder

As discussed in section 3.4, for processing the image input, a CNN architecture would be best. And training the model from scratch on various categories of images would be expensive and requires more computation. Therefore, an already trained model on ImageNet dataset which produced a very low top-5 error rate of 3.57% is used in this experiment i.e. ResNet 152. The working and other reasons for selecting ResNet 152 are already discussed in section 3.4.

Implementation steps related to encoder are discussed below.

- ResNet 152, which is already available in PyTorch '**torchvision**' library is used along with pre-trained weights for this experiment.
- As the goal is not to classify the image, instead encoding of the image is required, the *last 2 layers (Pooling and linear layer) of ResNet-152 model are removed/discarded.*
- A new '**AdaptiveAvgPool2d ()**' layer is introduced such that the encoding obtained from the encoder is resized to a fixed size. This helps to feed variable-length images to the Encoder. In this experiment, the images are resized to 256*256
- Since we are using 'Flickr30k' dataset and the ResNet152 was trained on ImageNet dataset, we may need to fine-tune the Encoder to get adopted for Image captioning task. To achieve this a '**fine_tune ()**' method is used, where this method enables or disables gradient calculations for parameters related to Encoder. Only the convolutional blocks from 2 to 4 of ResNet is used for the fine-tuning process, this is because the initial convolution block would have already learned certain fundamentals of the image.
- Two Encoder model classes are created, one for Global attention purpose and another for Adaptive attention purpose.
- The class for Global attention purpose uses all the steps discussed above
- Whereas the class used for Adaptive attention purpose has 2 extra layers added to the layers of Global attention i.e. **AvgPool2d ()** and a **linear layer**, this helps in representing spatial CNN features.

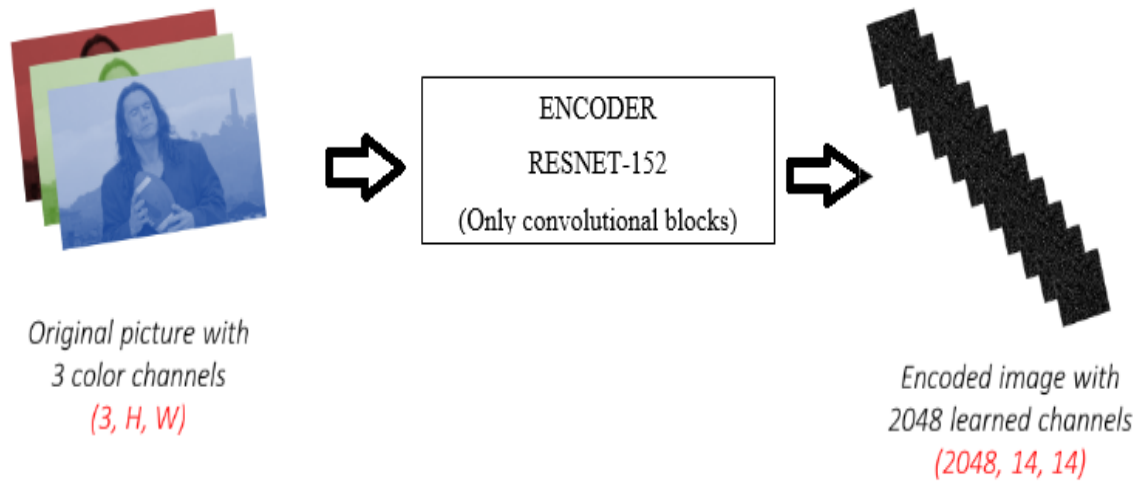


Figure 4.1: Image processing using ResNet-152 as an encoder.

The pre-processed image with 256×256 dimensions along with 3 colour channels is passed to an Encoder having ResNet-152 model as shown in Figure 4.1. The encoder was not trained from scratch in this experiment, instead, a concept of Transfer learning is used, where an already-trained network of ResNet-152 is present. The Encoder encodes the image and produces smaller representations of an image with 2048 learned channels. The combined representation of these channels provides certain useful information of the original image. The final size produced by the encoder is 14×14 with 2048 channels. In terms of Tensor, it can be represented as $2048 \times 14 \times 14$. This encoded image information produced by the encoder is used by attention mechanism to produce a calculated weight (score).

4.3 Global attention and Adaptive attention

As the goal of this experiment is to compare the Image captioning task using two attention mechanisms (Global and Adaptive attention), two attention models are created. Which are used respectively by the decoder model to generate captions.

Attention mechanism acts as a bridge between encoder and decoder, where the **encoded image** from learned channels of Encoder along with the **previous output of the decoder** is used to make decoder decide which part of the image needs to be paid importance for generating the next word. Figure 4.2 shows how the attention

mechanism uses encoder output and previous decoder output to generate a context vector, which helps the decoder to pay attention across the image.

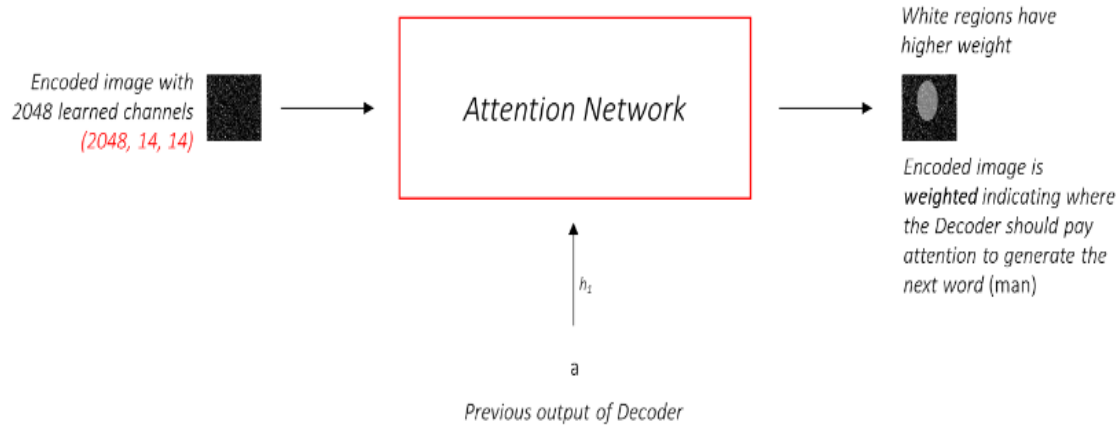


Figure 4.2: Attention mechanism to help the decoder to decide the next word of the caption.

Different attention models practise different mechanisms in indicating weighted vector. In this experiment as discussed in previous sections Global and Adaptive attentions are used.

The **Global attention model** is a simple combination of linear layers and a couple of activations.

- The first linear layer is used to transform the **encoded image**. Attention dimension of 512 is used in this experiment.
- The second linear layer also uses the attention dimension of 512 and transforms the **decoders previous output**.
- A third linear layer is used to transform the output of previous linear layers to a dimension of 1.
- Then it is followed by a ReLU activation function.
- Finally, a SoftMax function is used to generate the calculated weights, which is used by the decoder.

Adaptive attention follows an approach similar to global attention but contains two linear layers, Dropout and an activation function.

- The first and second linear layers use the attention dimension of 512 and transform the encoded image and decoders previous output to the same dimension.
- It is followed by a Dropout layer; this helps the next layer to randomly subsample and the capacity of the network is reduced.
- Finally, a softmax function is used to generate the calculated weights, which can be used by the decoder.

Thus, by following the above steps, the attention mechanism using the encoded image information and the Decoders previous output generates a calculated weight (can be called as ‘score’). Which plays a prominent role in the decoder to decide the next word to be generated.

4.4 Decoder (LSTM) with an attention mechanism

When an attention mechanism is used, the Decoder needs to look at different locations of the pixels based on the score (calculated weight) provided by the attention model to generate a word.

The Decoder uses an encoded image from the Encoder along with the calculated weight (score) obtained from attention layer to generate the word. Figure 4.3 provides a better understanding, where a decoder uses a weighted average across pixels of the image and is combined with a previously generated word to generate a new word.

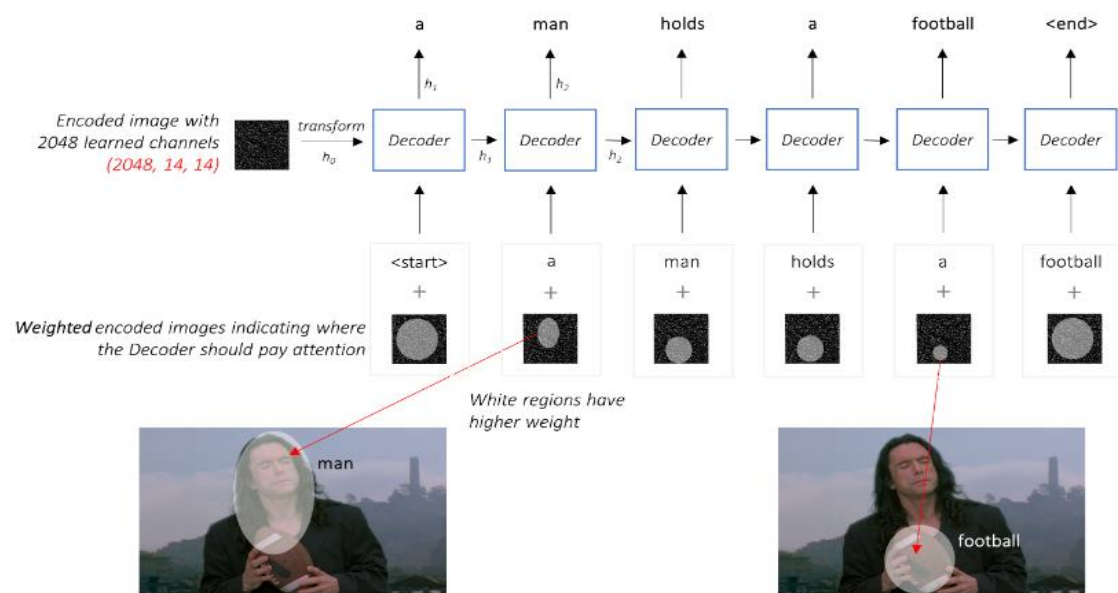


Figure 4.3: Process of generating captions by decoder using attention.

To achieve this, the implementations performed related to the decoder are as below.

- The encoder output is flattened to dimensions of $B, 14 \times 14, 2048$, where B is batch size.
- A method `init_hidden_state()` is created, where the encoded image is initialized to the hidden state and cell state of LSTM.
- The caption lengths are decreased and sorting of images available in the batch and captions is done. This helps to process only valid captions and not the extra `<pad>` which were inserted in the pre-processing stage.
- Only the effective regions at respective timestep are processed and this process is iterated for each of the timesteps.
- For using the attention at each decoder step, a PyTorch LSTMCell is used. LSTMCell is an operation provided by PyTorch to operate at single timestep.
- The Attention encoding from attention layer is passed to the decoder using a gate. And the gate used here is a sigmoid gate which is activated by a linear layer used for transforming the previous hidden state of the decoder.
- At each timestep, the decoder weights and attention-related weights are computed using an attention network.
- The obtained attention weight is concatenated with previous word embedding and an LSTMCell operation is executed such that an output is generated.
- A score for each word in the word map/corpus is generated using a linear layer, where it transforms output from LSTMCell to a score.
- And using these scores the next word to be generated is obtained.

4.5 Overall architecture and Training process

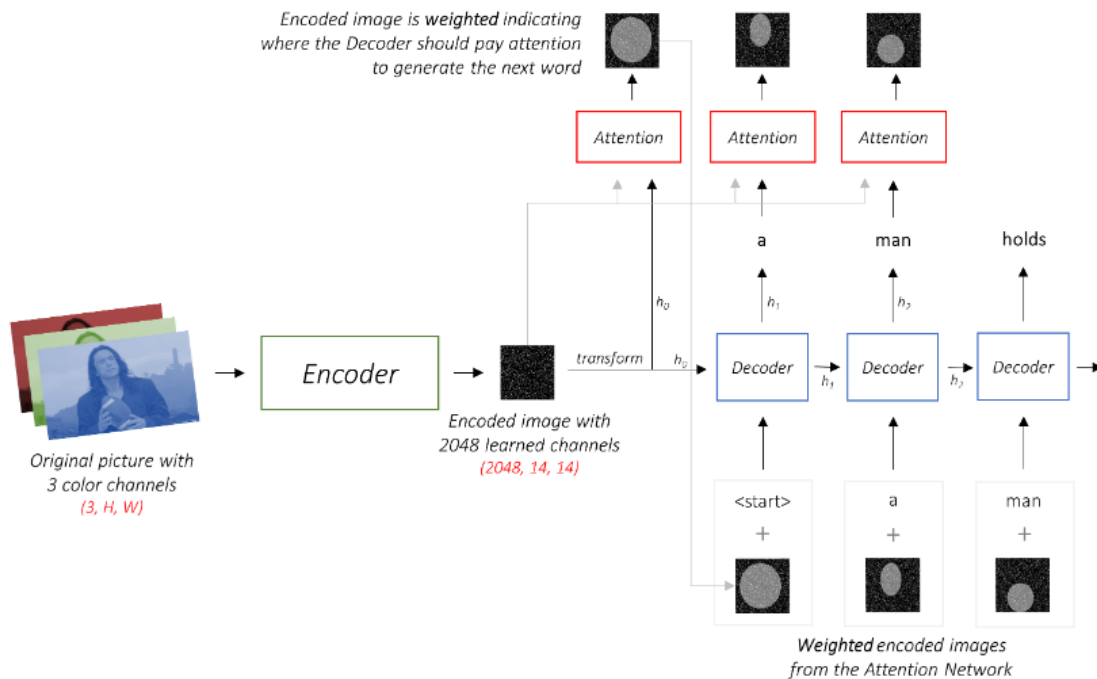


Figure 4.4: Overall model architecture for Image captioning.

The Implementation of Encoder, Decoder and attention mechanism has been discussed in Sections 4.1 to 4.4. The Overall architecture used in this architecture is a combination of these sections. Figure 4.4 provides an overview of the whole architecture.

- The encoder generates an encoding of the image and this forms the initial hidden state and cell state of LSTM decoder.
- At each timestep of decoder, attention network calculates the weights for each pixel of image using encoded image and decoders previous hidden state. The obtained weighted average from attention along with word which was previously generated is passed to LSTM for generating a new word.
- The obtained decoder output is transformed into a score for each word in the word map using a linear layer.

The model generates probability for each word in the vocabulary and using these probabilities the decoder produces a final sequence of words (sentence). In this

experiment there are 23457 words in the vocabulary and at each time step the probabilities of these words are obtained and decoder can't pick the highest probability word blindly, instead, it should concentrate on the likelihood of sentence.

Generally, there exist certain search algorithms, which considers the vocabulary and generates one or more approximation of decoder sentences for the available prediction.

In this experiment, one such search method called **Beam Search** is used.

Beam search decoder:

Using the available vocabulary and decoder probability, beam search estimates all possible words which may arise next and stores only 'k' likely words, where k is specified by the user. And once an end of sequence word is encountered the process of searching gets stopped. In this experiment, the captions are generated by using beam size 3 and 5.

Training process:

- Adam is used as an optimizer for both Encoder and decoder. In the encoder, it is used for fine-tuning.
- A very small learning rate of '**1e-4**' is used in encoder, this is because we are using already trained encoder and is quite optimized.
And for the decoder, an initial learning rate of '**4e-4**' is used.
- A '**CrossEntropyloss**' function is used as a Loss function for the overall model. The raw scores from the decoder are used and respective log and softmax operations are performed by the loss function.
- PyTorch's **pack_padded_sequence()** function is used to neglect the padded regions of the vector so that loss is not computed over these regions.
- The concept of **Teacher Forcing** is used in the training process. This is because by using this the model achieves faster convergence. Here instead of passing the previously generated decoder output as input in next time step, the original captions are passed as input at each timestep of the decoder.
- BLEU evaluation metric is used on validation dataset to evaluate model performance by using generated captions and reference captions. If the BLEU score starts degrading, then the training process is stopped. This is implemented because [Xu et al. \(2015\)](#) observed that after a certain point the correlation between BLEU score and loss gets a break.

- A batch size of 32 is used for training purpose. This is because the model is larger and contains gradients of the encoder and also fine-tuning is performed.

The overall model is trained two times using the above steps. First by using the Global attention mechanism and second by using the Adaptive attention mechanism.

Each approach is trained with respective hyperparameters and is trained for 25 epochs. *The training of each epoch took around 70-75 minutes.*

4.6 Results

The outputs of the experiment carried out is described in this section. This experiment was performed to determine if the use of Adaptive attention mechanism in Encoder-Decoder framework yields better evaluation metrics when compared with the usage of Global attention mechanism. The performance of the model on validation dataset is used to answer this question. Same training procedures, parameters and the dataset is used for both the models involving Adaptive attention and Global attention.

The Flickr30k dataset is having 31783 images which are generally related to human activities and using the karpathy split the dataset was split into 28k images for training, 1k images respectively for validation and test purpose. Same split and data are used in the experiments related to Adaptive and Global attention models.

Two models with a similar architecture having Adaptive attention and Global attention was created and trained with the procedures discussed in Section 4.5. And fine-tuned the respective models using Adam optimizer and CrossEntropyloss function.

4.6.1 Quantitative results

To find the performance of models on validation dataset BLEU, METEOR, ROUGE and CIDEr evaluation metrics are calculated with Beam size of 3 and Beam size 5 and the outcomes is recorded in Table 1 and Table 2.

Beam size 3 represents that three words are generated which are likely to be predicted to form a caption. And one word among these 3 which best suited for the caption is considered for generating the caption by the decoder.

Table 1: Performance evaluation of various Attention mechanisms on Flickr30k dataset with Beam size=3.

Attention method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE	CIDER
Global Attention	0.648	0.465	0.366	0.235	0.211	0.486	0.688
Adaptive Attention	0.662	0.482	0.387	0.252	0.246	0.517	0.706

Table 1 clearly shows that with a beam size of 3, the scores of all the seven evaluation metrics got significantly increased by using Adaptive attention mechanism instead of Global attention mechanism.

Similarly, the performance of models on Flickr30k dataset using the beam size of 5 is recorded in Table 2. And the results show that even with a beam size of 5, the Adaptive attention model outperforms in all the seven evaluation metrics when compared to Global attention mechanism.

Table 2: Performance evaluation of various Attention mechanisms on Flickr30k dataset with Beam size=5.

Attention method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE	CIDER
Global Attention	0.652	0.477	0.381	0.249	0.232	0.512	0.714
Adaptive Attention	0.681	0.496	0.406	0.268	0.257	0.558	0.761

Both the models are trained with the same parameters and with same procedures. But the performance of the model using the Adaptive attention mechanism is significantly high compared to the model with Global attention. One of the possible reasons would be that Global attention always focuses the image at each timestep irrespective of a non-visual word. But in case of Adaptive attention, the decoder doesn't focus image when any non-visual word is encountered, instead, it focuses on language model, which improved its score.

It is also observed from the tables that with the increase in Beam size the performance of both the models got increased. This is because the models received enough likely words to form a meaningful caption.

4.6.2 Qualitative results

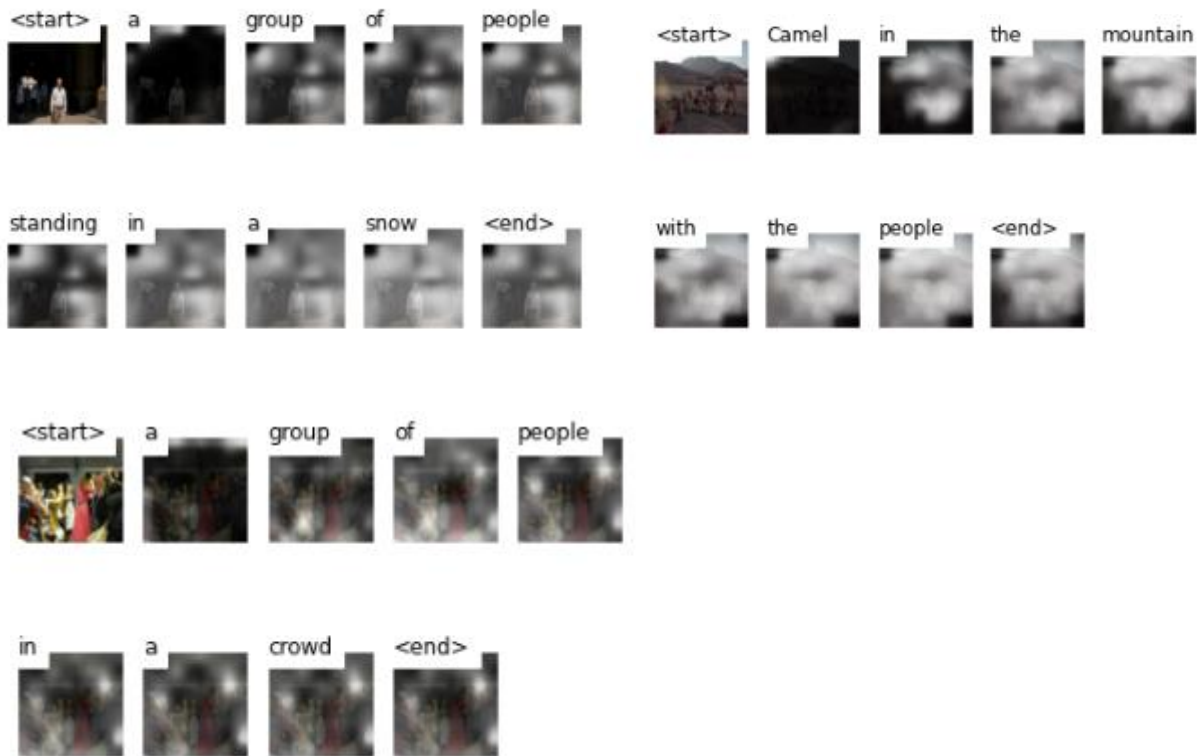





Figure 4.5: Words generated at each timestep of the decoder corresponding to the semantic meaning of the image block.

In the previous section, the quantitative results were discussed. And in this section, the qualitative output generated by the models are discussed.

Figure 4.5 contains the words generated at each timestep of the decoder process. This clearly shows that the focused region in the images was considered by the decoder in generating the next word. Also, it is observed that the semantic relationship exists between the focused region of the image and the word generated in the next timestep.

Some of the captions generated by the respective models along with the ground truth captions are as shown in below Table 3.

<p>1</p>		<p>Ground truth: an old man walking with a folder in his hand</p> <p>Global Attention: a group of people standing in a snow</p> <p>Adaptive Attention: a man in a white shirt and black pant is walking</p>
<p>2</p>		<p>Ground truth: people are dancing and clapping their hands.</p> <p>Global Attention: A group of people in a crowd.</p> <p>Adaptive Attention: People standing under the roof are clapping.</p>
<p>3</p>		<p>Ground Truth: a man with a gray beard sitting outside on a plastic storage crate.</p> <p>Global Attention: A man in a white shirt with a hat.</p> <p>Adaptive Attention: A man in a white shirt is sitting.</p>




4		<p>Ground truth: two people wearing jackets are watching a man set up a window display in a store</p> <p>Global Attention: People in a store in a jacket watching.</p> <p>Adaptive Attention: two people display window with a man.</p>
5		<p>Ground truth: People and camels taking a break in the desert</p> <p>Global Attention: Camel in the mountain with the people</p> <p>Adaptive Attention: A group of camel is sitting in snow</p>
6		<p>Ground truth: a woman in a leopard dress taking a picture with her cellphone.</p> <p>Global Attention: A woman wearing a leopard using cellphone.</p> <p>Adaptive Attention: A brown hair with a cellphone in hand.</p>

Table 3: Captions generated by Global and Adaptive attention models.

As discussed in the dataset each image contains 5 ground truth captions, which was collected using crowdsourcing.

The captions generated by the respective models are recorded in Table 3, where a single ground truth caption is displayed. This is because each image contains 5 ground truth caption and the captions generated by the models can be closer to any of these ground truth caption or certain phrases of these captions or would be entirely different from the ground truth captions. Therefore, the ground truth caption which is almost similar to generated captions alone is used in the table.

A sample of 6 images from validation dataset was selected and passed the same image to both the attention models and the obtained captions were recorded and updated in the above table.

The captions generated by both the models were satisfactory, where none of the models was able to provide a perfect essence of the image. This may be due to many reasons like, the shortage of images related to a particular category in the dataset, which resulted in the bias towards a particular category of images.

By observing the images 1,2,4,5 we see that both the models can detect a group of peoples/objects. This may be due to the availability of many images in the training process where more than one people is present. Due to which the models were able to identify the individual and group of people.

Also, it is observed that none of the captions generated by both models is completely out of sense. The models were able to observe at least partial essence of the image, which helped them to detect the objects in the image and construct a caption.

The usage of a bigger dataset for training purpose may help models to grasp the higher essence of the image.

By observing the image 4 from the table, we can see that the captions generated for the longer ground truth sentence are not meaningful, this may be because the dataset used to train the models may not have enough categories of information and also not many captions with longer sequences.

Flickr30k is a dataset which contains entirely human-related activities. Therefore, to test the model's performance on a completely unknown set of categories of data, an animal image is passed to the model and below caption was generated



Global Attention: A group of people are walking through the snow
Adaptive Attention: A man in a blue shirt is riding a bike



This output clearly shows that the model was trained only with the few categories of data which are available in Flickr30k dataset. And the models don't provide a better caption for the images which are not present in the training dataset.

5. EVALUATION AND DISCUSSION

The various procedures used for evaluating the model is discussed in detail along with the strengths and limitations of the results in this section.

5.1 Evaluation of Results

After the process of training is completed, the generation of captions to the image and evaluating the model need to be done to examine the performance of the model. In this section, the processes involved in evaluating the model is discussed in detail.

- The concept of **Teacher Forcing** was used in Decoder during the training process; where the original image caption which was available from the dataset was used as input at each timestep of the decoder. But for validating the model, we need to use the word which is generated by the decoder at the previous time step as input to a current timestep of the decoder.
- To achieve the above step, a function named '**caption_image_beam_search()**' is created. This function encodes the input image and generates the possible word which needs to be appeared next in the caption by applying a beam search algorithm and is passed as input to the decoders current timestep.
- The generated/predicted caption from the decoder for all the images present in validation dataset is used in determining the model performance.
- The evaluation metrics like BLEU, METEOR, ROUGE and CIDEr scores are calculated using the decoder generated caption and the original image caption.
- This provides the overall evaluation scores representing the model performance.
- A function called '**visualize_att**' is created to visualize the words generated at each timestep of the decoder along with the calculated weights.

5.2 Strengths of Result

- Observation of results shows that the usage of deep CNN model (ResNet-152) impacted the final results and provides confidence to use deeper architecture greater than 152 layers.
- The use of Flickr30k dataset made the model to categorize many objects and helped in identifying and generating meaningful captions, this provided a hope to use datasets like MSCOCO having 330k+ images. Increase in the dataset can make the model more generalized and identify many objects.
- Even after training models with only 25 epochs, the captions generated are meaningful and satisfactory, training the model further provides hope of generating more meaningful and satisfactory captions.
- While training the model, BLEU evaluation metric is used to judge the quality of the model. This helped models to early stop the training by observing the BLEU metric score. This provides the hope to perform hyperparameter tuning as the score degrades.

5.3 Limitations of Result

The high validation score was not able to achieve by the models because of the constraints like

- Using an encoder model which was already trained on a different dataset. Training an encoder from scratch requires huge computation and a rich dataset to identify various categories of data. With the current computational resource, it was not possible.
- Availability of good GPU and the vast dataset was another constraint if availed would have made the models more generalized and trained for longer time
- Training the model with a greater number of epochs would have resulted in a better generation of the caption.
- If time permitted, other models like VGG, Inception could have been used as Encoder and tested.

- Also, the word embeddings were learned from scratch in this experiment, instead, an already well-trained word embeddings like GloVe, CBOW, skip-gram, etc. could have been used and tested.
- Other attention models which use the semantics of image and other optimization strategies could have been used to compare with the Adaptive and Global attentions.
- There is no perfect evaluation metric available to calculate the performance of Image captioning tasks. All the metrics were initially developed for the task of Machine Translation. Even though the metric shows a lower score, the model can produce a meaningful caption using the essence of the image.

6. CONCLUSION AND FUTURE WORK

The Evaluation of results along with strengths and limitations were discussed in Chapter 5. The overview of the whole processes followed to achieve the experiment and the future works which can be used to improve the performance is discussed in this section.

6.1 Overview of Research and Experiment

This research targeted a fundamental problem in the field of Artificial Intelligence, where the combination of two giant research fields of Computer vision and Natural language processing is involved. Based on successes of various attention mechanisms in the field of Natural language processing, two state-of-the-art attention mechanisms which showed a promising result was used in achieving the task of Image Captioning. The implementation and comparison of these attention mechanisms in generating meaningful captions are recorded in this experiment.

To experiment, the Flickr30k dataset was used. And the dataset was split using the split proposed by [Karpathy & Fei-Fei \(2015\)](#), which contained overall 28k images for training, 1k for validation and test respectively. Initially, the dataset was pre-processed, where the images were resized and transformed to fit in the encoder and the respective captions of images were processed to achieve the training process. During the training process, the dataset was passed to Encoder to generate the encoded information of the image and using the respective Attention mechanism the calculated weight (scores) of each pixel is generated, this helps the decoder to decide which part of the image needs to be focussed while generating the caption for the image. The decoder uses Beam search to estimate the possible words which can be generated in the caption. The results are evaluated using the BLEU, METEOR, ROUGE and CIDEr metrics and the results obtained are recorded. The captions generated by the respective Adaptive and Global attention models are shown in Figure 4.6.

The overall results show that the model using Adaptive attention has shown a better performance than the model using Global attention.

6.2 Contributions and Impact

A lot of active research is currently undergoing in the fields of Natural language processing and Computer vision. And by combining these two fields, provides novel approaches and processes to handle the tasks which later can be used in respective fields and other fields. Also, it can be combined with the field of Audio processing, where the generated captions can be used by applications like Alexa, Siri etc to speak, so that the persons who are illiterate and cannot read and write can understand the essence of image/video.

Apart from a research perspective, the task of Image captioning plays an important role in developing applications which can assist visually impaired persons in understanding the surrounding, in tasks relate to scene understanding; where the hidden objects in the image/video are recorded and can be used by investigative agencies ([Wang, Zhang, & Yu, 2020](#)), in the automatic creation of metadata for images (indexing) for use by search engines, in general-purpose robot vision systems.

Also, in the field of education, where it helps children in understanding various objects/information across the world, in various teaching institutes to discuss the essence of images, in tasks related to question and answering etc.

Many other fields where the involvement of Computer vision, natural language processing, audio processing can also benefit from this kind of research.

6.3 Future Work & Recommendations

- This research was performed on captioning the images. Similarly, using the concepts discussed in this experiment captioning can be performed on videos.

- Using various attention mechanisms in advanced architectures like BERT and Transformers can be used in generating the captions.
- Instead of learning word embeddings from scratch, pre-trained word embeddings like GloVe, CBOW, skip-gram, etc. can be used.
- Usage of Generative Adversarial Network (GAN) along with Reinforcement learning can experiment. This helps in solving the problem of exposure bias in supervised training where RNN/ LSTM based architectures are involved.

BIBLIOGRAPHY

- Amirian, S., Rasheed, K., Taha, T. R., & Arabnia, H. R. (2019). A short review on image caption generation with deep learning. In *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV)* (pp. 10-18). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).
- 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).
- Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.
- Brownlee, J. (2019, July 05). A Gentle Introduction to the ImageNet Challenge (ILSVRC). Retrieved from <https://machinelearningmastery.com/introduction-to-the-imagenet-large-scale-visual-recognition-challenge-ilstvrc/>
- Chan, W., Jaitly, N., Le, Q., & Vinyals, O. (2016, March). Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 4960-4964). IEEE.
- Chaudhari, S., Polatkan, G., Ramanath, R., & Mithal, V. (2019). An attentive survey of attention models. *arXiv preprint arXiv:1904.02874*.
- Chen, X., & Zitnick, C. L. (2014). Learning a recurrent visual representation for image caption generation. *arXiv preprint arXiv:1411.5654*.
- Cho, K., Courville, A., & Bengio, Y. (2015). Describing multimedia content using attention-based encoder-decoder networks. *IEEE Transactions on Multimedia*, 17(11), 1875-1886.
- Ciresan, D., Giusti, A., Gambardella, L., & Schmidhuber, J. (2012). Deep neural networks segment neuronal membranes in electron microscopy images. *Advances in neural information processing systems*, 25, 2843-2851.
- Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition* (pp. 248-255). Ieee.
- Dwivedi, P. (2020). Understanding and Coding a ResNet in Keras. Retrieved from <https://towardsdatascience.com/understanding-and-coding-a-resnet-in-keras-446d7ff84d33>
- 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.
- Galassi, A., Lippi, M., & Torroni, P. (2020). Attention in Natural Language Processing. *IEEE Transactions on Neural Networks and Learning Systems*.

- Gershgorn, D., 2020. *The Data That Transformed AI Research—And Possibly The World*. Retrieved from <https://qz.com/1034972/the-data-that-changed-the-direction-of-ai-research-and-possibly-the-world/>
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).
- Hendricks, L. A., Venugopalan, S., Rohrbach, M., Mooney, R., Saenko, K., & Darrell, T. (2016). Deep compositional captioning: Describing novel object categories without paired training data. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-10).
- Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.
- Hochreiter, S., Bengio, Y., Frasconi, P., & Schmidhuber, J. (2001). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies.
- 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).
- Khan, A., Sohail, A., Zahoor, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, 53(8), 5455-5516.
- Kiela, D., Wang, C., & Cho, K. (2018). Dynamic meta-embeddings for improved sentence representations. *arXiv preprint arXiv:1804.07983*.
- Kiros, R., Salakhutdinov, R., & Zemel, R. (2014, January). Multimodal neural language models. In *International conference on machine learning* (pp. 595-603).
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84-90.
- Kulkarni, G., Premraj, V., Dhar, S., Li, S., Choi, Y., Berg, A. C., & Berg, T. L. (2011). Baby talk: Understanding and generating image descriptions Proceedings of the 24th CVPR.
- Kuznetsova, P., Ordonez, V., Berg, T. L., & Choi, Y. (2014). Treetalk: Composition and compression of trees for image descriptions. *Transactions of the Association for Computational Linguistics*, 2, 351-362.
- Lebret, R., Pinheiro, P. O., & Collobert, R. (2015). Phrase-based image captioning. *arXiv preprint arXiv:1502.03671*.
- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4), 541-551.
- LeCun, Y., Jackel, L. D., Bottou, L., Cortes, C., Denker, J. S., Drucker, H., ... & Vapnik, V. (1995). Learning algorithms for classification: A comparison on handwritten digit recognition. *Neural networks: the statistical mechanics perspective*, 261, 276.

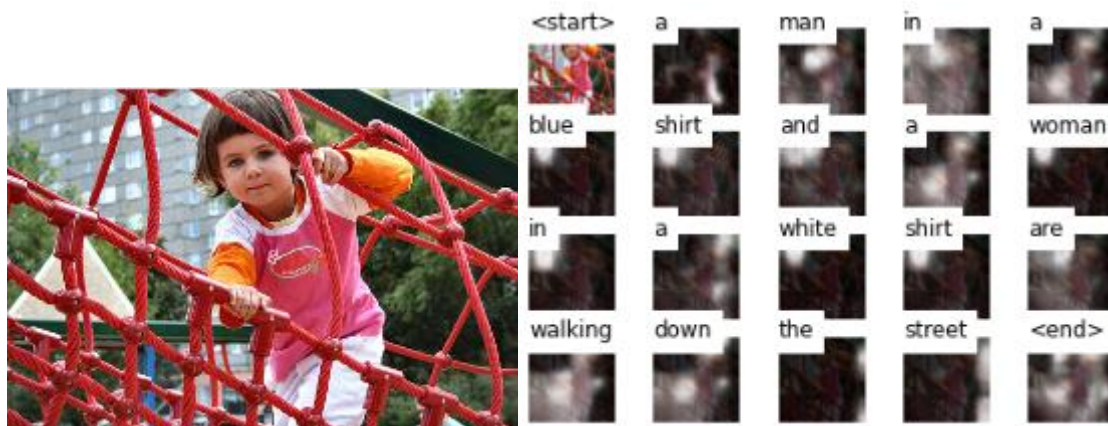
- Lin, Z., Feng, M., Santos, C. N. D., Yu, M., Xiang, B., Zhou, B., & Bengio, Y. (2017). A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130*.
- Liu, X., Deng, Z., & Yang, Y. (2019). Recent progress in semantic image segmentation. *Artificial Intelligence Review*, 52(2), 1089-1106.
- Lu, J., Xiong, C., Parikh, D., & Socher, R. (2017). Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 375-383).
- Lu, J., Yang, J., Batra, D., & Parikh, D. (2016). Hierarchical question-image co-attention for visual question answering. *Advances in neural information processing systems*, 29, 289-297.
- Luong, M. T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.
- Maharjan, S., Montes, M., González, F. A., & Solorio, T. (2018). A genre-aware attention model to improve the likability prediction of books. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (pp. 3381-3391).
- Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z., & Yuille, A. (2014). Deep captioning with multimodal recurrent neural networks (m-rnn). *arXiv preprint arXiv:1412.6632*.
- Mason, R., & Charniak, E. (2014, June). Nonparametric method for data-driven image captioning. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (pp. 592-598).
- Moore, G. E. (1965). Cramming more components onto integrated circuits. *Electronics*, 38(8). New York: Mcgraw-Hill.
- Olah, C. (2015). Understanding LSTM Networks -- colah's blog. Retrieved from <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>
- Plummer, B. A., Wang, L., Cervantes, C. M., Caicedo, J. C., Hockenmaier, J., & Lazebnik, S. (2015). Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision* (pp. 2641-2649).
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3), 211-252.
- Schmidhuber, J., & Hochreiter, S. (1997). Long short-term memory. *Neural Comput*, 9(8), 1735-1780.
- Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., & LeCun, Y. (2013). Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*.
- Shen, T., Zhou, T., Long, G., Jiang, J., Pan, S., & Zhang, C. (2017). Disan: Directional self-attention network for rnn/cnn-free language understanding. *arXiv preprint arXiv:1709.04696*.
- Simonyan K, Zisserman A (2015) VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION. ICLR 75:398–406. doi: 10.2146/ajhp170251.

- Staniūtė, R., & Šešok, D. (2019). A systematic literature review on image captioning. *Applied Sciences*, 9(10), 2024.
- Sukhbaatar, S., Weston, J., & Fergus, R. (2015). End-to-end memory networks. *Advances in neural information processing systems*, 28, 2440-2448.
- Szegedy C, Ioffe S, Vanhoucke V (2016a) Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. arXiv Prepr arXiv160207261v2 131:262–263. doi: 10.1007/s10236-015-0809-y.
- Szegedy C, Vanhoucke V, Ioffe S, et al (2016b) Rethinking the Inception Architecture for Computer Vision. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE, pp 2818–2826.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998-6008).
- Vinyals, O., Fortunato, M., & Jaitly, N. (2015). Pointer networks. *Advances in neural information processing systems*, 28, 2692-2700.
- Wang, F., & Tax, D. M. (2016). Survey on the attention based RNN model and its applications in computer vision. *arXiv preprint arXiv:1601.06823*.
- Wang, H., Zhang, Y., & Yu, X. (2020). An Overview of Image Caption Generation Methods. *Computational Intelligence and Neuroscience*, 2020.
- Wang, J., Cao, Z., Xiao, Y., & Qi, X. (2018, March). Supervised guiding long-short term memory for image caption generation based on object classes. In *MIPPR 2017: Pattern Recognition and Computer Vision* (Vol. 10609, p. 106090P). International Society for Optics and Photonics.
- Wang, Y., Huang, M., Zhu, X., & Zhao, L. (2016, November). Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing* (pp. 606-615).
- Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., ... & Bengio, Y. (2015, June). Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning* (pp. 2048-2057).
- Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016, June). Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies* (pp. 1480-1489).
- Yang, Z., Yuan, Y., Wu, Y., Salakhudinov, R., & Cohen, W. W. Encode, review, and decode: Reviewer module for caption generation. arxiv 2016. *arXiv preprint arXiv:1605.07912*.
- Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., & Courville, A. (2015). Describing videos by exploiting temporal structure. In *Proceedings of the IEEE international conference on computer vision* (pp. 4507-4515).

- You, Q., Jin, H., Wang, Z., Fang, C., & Luo, J. (2016). Image captioning with semantic attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4651-4659).
- Young, P., Lai, A., Hodosh, M., & Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2, 67-78.
- Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In *European conference on computer vision* (pp. 818-833). Springer, Cham.
- Zhao, S., & Zhang, Z. (2018, January). Attention-via-attention neural machine translation. In *AAAI*.

APPENDIX A

Bad captions generated by the Global Attention model



Bad captions generated by the Adaptive attention model



<start> a man in a
blue shirt and a hat
is standing in front of
a mountain <end>



<start> a little boy is
playing in the snow <end>