2021

Evaluating the Performance of Transformer architecture over Attention architecture on Image Captioning

Deepti Balasubramaniam
Technological University Dublin

Evaluating the Performance of Transformer architecture over Attention architecture on Image Captioning

Deepti Balasubramaniam

D20123887

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

I certify that this dissertation which I now submit for examination for the award of MSc in Computing(Data Science), is entirely my own 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 test 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.

Signed: 

Date: **05 January 2022**

ABSTRACT

Over the last few decades computer vision and Natural Language processing has shown tremendous improvement in different tasks such as image captioning, video captioning, machine translation etc using deep learning models. However, there were not much researches related to image captioning based on transformers and how it outperforms other models that were implemented for image captioning. In this study will be designing a simple encoder-decoder model, attention model and transformer model for image captioning using Flickr8K dataset where will be discussing about the hyperparameters of the model, type of pre-trained model used and how long the model has been trained. Furthermore, will be comparing the captions generated by attention model and transformer model using BLEU score metrics, which will be further analysed using human evaluation conducted using intrinsic approach. After analysis of results obtained using statistical test conducted on BLEU score metrics and human evaluation it was found that transformer model with multi-head attention has outperformed attention model in image captioning.

Key words: *computer vision, natural language processing, deep learning, encoder-decoder model, transformer model, attention mode, BLEU metrics*

ACKNOWLEDGEMENTS

I would like to express my sincere thanks to Dr. Robert Ross for his assistance, support, motivation and patience throughout this MSc dissertation.

I would also like to express my sincere gratitude to Dr. Emma Murphy and Dr. Luca Longo for advising on different aspects about the dissertation.

Also, I would like to thank all of the professors in Technological University Dublin, who guided and supported me throughout my study in the University.

Last but not least, I would like to thank my family and friends who always encouraged and supported me during this MSc.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

1. INTRODUCTION

1.1 Background

Image captioning is the process which generates textual description for the given set of images. Image captioning can be used in various area like medicine, military, automatic cars, digital library, web searching etc. There are various methods in which image captioning can be implemented. The recent advancement in Artificial Intelligence has greatly improved the performance of the models. However, it is difficult for ,machines to imitate human brain and models cannot be the exact replica of ground truth. This research will be focusing on deep learning methods through which image captioning can be implemented. Few researchers have already implemented the methods like merged encoder-decoder and attention model. In this method will be using transformers with multi-head attention which is a novice method and will be comparing with already implemented methods.

Now, in general image captioning works based on concept of sequence to sequence problem, where images are regarded as series or sequence of pixels and gets converted to sequence of words. Processing of both images and words/sentences need to be considered (Roy, 2020). Convolution Neural network (CNN) is used for image part and Recurrent Neural Network (RNN) is used for language part. Different architecture needs to be implemented to know in which order the piece of information should be introduced to the mode. There are two types of architecture such as injecting architecture and merging architecture (Roy, 2020). In injecting architecture RNN is trained on the mixture of image and language data, where both the data are introduced together at the same time. In contrast, multimodal layer architecture is created in merging architecture where image and language data are encoded separately and added to a feed-forward network. Another, enhanced architecture used is encoder-decoder with attention, when each word of the caption produced by the sequence decoder, attention models helps to focus on the part of the image that is most related to the word it is generating (Doshi, 2021). Next, novel architecture used for image captioning is transformers, which is similar to encoder-decoder but replaced LSTM. In this research will be comparing the transformer model with the merged encoder-decoder and

encoder-decoder with attention model. The relevant code for this experiment can be seen in GitHub[1].

1.2 Research Project/problem

Merged encoder-decoder and attention mechanism are one most traditionally used approaches for image captioning using deep learning. But these models have some limitations, encoder-decoder architecture does not consider the spatial features and generates the caption as a whole, attention mechanism adds more weight parameters and increases the training time and also it has context vector with fixed-length and fails to retain longer sequences, therefore with the use of transformers this issue can be resolved as it allows for parallelization where input sentences are achieved in parallel and passes simultaneously all the words in the sentence.

1.3 Research Objectives

The study will be based on Natural language processing and computer vision. Throughout our research will be implementing three major methods of image captioning such as merged encoder-decoder model, encoder-decoder model with attention and transformers with multi-head attention and will be comparing these models. Will be starting with traditional approach of merged encoder-decoder model using Python where both image and text will be encoded together and will feed to decoder. In this research will be doing 'image model' with CNN and language model using RNN/LSTM through text sequences of varying length will be encoded. Transfer learning to encode the image features will be used. There are lot of pre-trained models that are available and can be used for image captioning such as ResNet, InceptionV3 and VGG-16. However for this research purpose will be using inception v3 which has least number of training parameters and can outperform other models. For encoding text sequence will be using pretrained Glove model such that each word is mapped to 200 dimensional vector. This mapping will be carried out in a separate layer called embedding layer. Greedy search and beam search are most popular methods used to generate the captions which will be used to pick the best word that could define the image accurately.

[1] https://github.com/DeeptiBSV/ImageCaptioning_Thesis

Number of datasets are used for training, testing and evaluation of image captioning methods. Based on the research purpose these datasets can be used. The most commonly used dataset for image captioning are MSCOCO, Flickr8K and Flickr30K.

In Flickr8K dataset , each of the image consists of five different captions that describes the events and objects in the image. For the starting phase Flickr8K dataset can be used as it is comparatively smaller in size than the other two datasets. For more advanced research Flickr30K and MSCOCO dataset can be used. Once finalizing the dataset need to start with the image caption generator code. Step1, will be importing required libraries like NumPy, keras, pyplot. In Step 2, it will be data loading and pre-processing where will be defining all the paths that need to save all the image ids and its captions. Now, need to create a dictionary such that key contains the name of the image and value stores all the 5 captions of the image. Following this step, to remove punctuations text cleaning is required and to convert all the words to lower case. Next need to create the vocabulary for all the unique words present in the training dataset. The image id and its new cleaned text needs to be stored in the same format. The training images needs to be loaded in train variable and will load the dictionary related to trained images followed by adding two tokens such as ‘startseq’ and ‘endseq’. Then, vocabulary size needs to be reduced such that words that occur at least 10 times in the corpus need to present in our model. In Step3, will be using Glove embedding for deriving semantic relationships between words from the cooccurrence matrix.Step4 will be model building and training, where will be opting transfer learning with the help of inceptionv3 network pretrained with ImageNet dataset. While designing the model, it needs to be merged , where sequences from the text needs to be processed, feature vector from the image must be extracted. And then using softmax the output is decoded by concatenating the two layers. The third layer of the input will be the partial caption of max length 34 which is fed in to the embedding layer and then these words are mapped to glove embedding of 200-dimension. To avoid overfitting will be using the dropout layer and will be fed to the LSTM for processing the sequence. Step 5 will be model training, during this step model will be trained with 20 or 30 epochs with 5 batch size. For training the model will be using colab GPU to avoid the computational power. Step 6 will be greedy and beam search. As the model would generate more than 1000 long vector, will be picking the word with highest probability greedily and this process is considered as Greedy Search method.

The second model which will be used for this research is attention mechanism for image captioning. The self-selection ability is called attention. Through attention mechanism model would able to pick specific features by focusing on the subset of inputs. Attention mechanism looks for salient features in the image instead of considering the entire image to a static representation. While building the model with attention mechanism will be following similar steps as followed for merged encoder-decoder model, however, during model definition will be using attention mechanism.

The third model which will be used for this research is transformer with multi-head attention model. The transformer model implements encoder-decoder architecture, the major difference is that transformer can parallelly receive input/output sequence without a time step. The encoder block will be having two-sub layers with first layer having multi-head attention mechanism and second layer have fully connected feed forward network. Contextual relationship between every word in a sentence can be achieved by the attention vector generated by every word. The multi-head attention layer over the output of encoder stack would be third layer. Will be implementing this logic of transformer for our image captioning model using TensorFlow. During this process will be repeating all three steps done for previous model, in step 4 will be doing positional encoding which uses sin and cosine functions of different frequencies. For every odd index on the input vector uses cos function to create a vector and uses sin function to create vector in every even index. These vectors will added together to their corresponding input embeddings. Step 5 will be multi-head attention function, where all the dimensions must be matched. Step 6 and 7 will be creating encoder - decoder and transformer class. In Step 8 will be defining the parameter for hyperparameter. In step 9 will train the model.

After completing all the model, final step would be the evaluation, using BLEU metrics to compare the scores of each model. Based on the scores will be able to come to the conclusion on which hypothesis to accept or reject.

1.4 Research Methodologies

Methodologies of the research has been clearly identified for the clear validation and reliability during the experiment.

The research is a secondary research method, as it is a synthesis of an existing research where research investigation started with the selection of appropriate dataset for the research followed by pre-processing, model definition and model training. The research falls under the category of mixed research method as it uses the combination of quantitative and qualitative methods for data collections and its analysis, also focus on the strengths of each approach and their different weaknesses.

The research follows empirical research method as it involves gaining knowledge by observing the data and involves in defining the hypothesis test and prediction. Deductive reasoning will be applied for this research as the research starts with hypothesis testing, supporting data evidence is provided in order to test the hypothesis and conclusion is drawn based on the analysis.

1.5 Scope Limitations and Delimitations

The aim of the research is to implement transformer model in python to test whether the model gives better BLEU score, when used in Flickr8K dataset based in image captioning than compared to attention model.

The entire research will be conducted using Flickr8k dataset, therefore the dataset size and its available values are not enough to come to a strong conclusion on the model performance. Also, for the score evaluation, BLEU metrics will be used which is more accurate for short texts and its increased score might not assure that model has performed well.

Exploring the text part in image captioning is only in the scope for this study, where all the analysis will be implemented using colab, the library that will be used are keras, transformers, pandas and numpy. VGG16 and Inception v3 are the pretrained Convolution Neural Network used for this study, other pre-trained models such as Xception , ResNet are not in this research scope. Dataset used will be Flickr8K, other datasets like MSCOCO or Flickr30k is beyond this scope. Furthermore, BLEU score metrics is used only for attention model and transformer model for the comparison between the model, BLEU score for simple encoder-decoder model is not in the scope as there are previous papers and researches who have already compared and proved that attention model is better than simple encoder-decoder model.

1.6 Document Outline

Chapter 2-Literature Review

This chapter is dedicated for the literature survey of the previous research papers and their proposals, which will help in knowing the evolution of the image captioning methods, lessons learnt and issues with the approach. With the help of such literature survey, one could able to implement the methods based on the proposals by other researches and enhance it further for the efficient performance or could propose a better solution for the related domain.

Chapter 3 – Design and Methodology

In this chapter has discussed about the proposed methods for the image captioning and provided insights to the experiment to test the hypothesis based on the results from each model. This chapter contains detailed explanation about the dataset, model and its output.

Chapter 4 – Results, Evaluation and Discussion

Detailed analysis of the results and output from each model is been discussed in this chapter. Suitable evaluation metrics has been obtained and compared using a statistical test in order to reject or accept the null hypothesis. Furthermore, have also discussed about the human evaluation and how it conducted and its results. Finally, from the output have come to the conclusion for the experiment.

Chapter 5 – Conclusion

In this chapter, have discussed and summarized about the overall analysis and results obtained from the experiment conducted through this research and have suggested the future scope for the research as an extension to the paper.

2. LITERATURE REVIEW

2.1 Template Based

In this section will be reviewing and describing few of the existing image captioning methods which includes different types like template based image captioning, retrieval based image captioning and novel caption generation (Hossain et al., 2019). In template based approaches different attributes, objects and actions are first detected and then the templates with blank spaces are filled (Hossain et al., 2019). Hutchison et al. (2010) have used this method in such a way that template slots are filled with triplet of scene elements for generating image captions. Li et al. (2011) used web based n-grams and computer vision based images for automatic image caption and did not use pre-existing text relevant to image. This approach consist of two steps n-gram phrase selection and phrase fusion which focuses on relationships between the extracted phrases related to objects and attributes. Kulkarni et al. (2013), proposed the method inferring the objects and attributes before filling the gaps.

2.2 Retrieval based

In retrieval based approaches, existing captions are used for retrieval of captions. Captions are generally retrieved from visual and multimodal space. In retrieval based methods, visually similar images are matched with their captions available from training dataset, which are generally called as candidate captions. Hodosh et al., 2013 proposed ranking method where pools of images were captured and system ranks the caption of that image over the captions of all other test images. Ordonez et al., 2011 proposes two extractive features for generation of image description. It select relevant captions, where it first uses global image representation and then incorporates features from estimates of image content .Novel captions can be generated from both multimodal and visual space. This approach in general analyses the image content first and then using language model generate the image content form the visual content. These methods are able to generate new captions for each image with semantically accurate than previous approaches (Chu et al., 2020). Image captioning based on deep

learning can be categorized in to three different learning techniques such as supervised learning technique, Unsupervised learning techniques and reinforcement learning technique. Most of the paper in general focuses on simple encoder-decoder architecture which in general used LSTM as language model or CNN-RNN or compositional architecture which used transformers along with encoder-decoder architecture.

2.3 Novel Image Captioning

2.3.1 Attention Based Image Captioning

Attention mechanism, obtained from the study of human vision which involves complex intellectual ability that human beings have in cognitive neurology (Wang et al., 2020). When information is received human beings tend to overlook the primary information while ignoring the secondary information, such capacity of self-selection is known as attention. This mechanism was first proposed during the study of image classification in the field of visual images which uses attention mechanism in RNN (Wang et al., 2020). In Natural Language processing, when humans read long texts, they tend to notice only the key words or entities. Luong et al. (2015), has proved that attention mechanism can be used on machine translation and on abstract generation by Rush et al. (2015) which achieved remarkable results. In Neural network models, attention mechanism allows the neural network to have the ability to focus on its subset of input or features (Wang et al., 2020). The major part of the attention mechanism would be to have two aspects, one aspect is to make decision on to pay attention to certain part of the input and the other aspect would be to assign limited information processing to the relevant part. At present calculation of attention mechanism formula are shown below as per author (Wang et al., 2020), which focuses on linking the target module m_t to source module m_s using a function and probability distribution is achieved by normalizing it.

$$a_t = \text{align}\left(m_t, m_s\right) = \frac{\exp(f(m_t, m_s))}{\sum_{s'} \exp(f(m_t, m_{s'}))}, \quad \text{(2.3.1-1)}$$

$$f(m_t, m_s) = \begin{cases} m_t^T m_s, & \text{dot}, \\ m_t^T W_a m_s, & \text{general}, \\ W_a [m_t; m_s], & \text{concat}, \\ v_a^T \tanh(W_a m_t + U_a m_s), & \text{perception}. \end{cases} \quad (2.3.1\text{-}2)$$

There are different types of attention mechanism algorithms, suggested by few authors which will be reviewed in this chapter.

- **Soft Attention**: Bahdanau et al., 2016 has first proposed the idea of soft attention on machine translation, where soft term refers to probability distribution of attention distribution. The probability is given based on context vector Z_t for any input sentence S. Finally, the probability distribution is achieved by calculating the weighted sum of all region.

$$E_{p(s_t|a)}[\hat{z}_t] = \sum_{i=1}^{L} \alpha_{t,i} a_i . \quad (2.3.1\text{-}3)$$

  According to Bahdanau et al., 2016 a deterministic attention model is formulated by computing a soft attention weighted attetion vector (Bahdanau et al., 2016 ; Wang et al., 2020).

$$L = -\log(P(y|x)) + \lambda \sum_i^L \left(1 - \sum_t^C \alpha_{t,i}\right)^2 . \quad (2.3.1\text{-}4)$$

- **Hard Attention**: Hard attention focus on only one location and selects the unique location randomly, which is unlike soft attention where weighted sum of all regions are calculated. In hard attention instead of sampling the entire encoder, it uses probability on hidden state of input. According to Xu et al., 2016 context vector Z_t vector is calculated as below. Monte Carlo sampling is needed to achieve gradient back propagation. Hard attention selects the information based on random sampling or method of maximum sampling, therefore it is difficult to achieve the functionality difference between the attention distribution and final loss function (Wang et al., 2020).

$$p(s_{t,i} = 1 | a) = \alpha_{t,i},$$
$$\hat{z}_t = \sum_{i=1}^{L} s_{t,i} a_i , \quad (2.3.1\text{-}5)$$

- **Multihead Attention :** Input information generally represented in key-value pair format, in which attention distribution is considered as “key” and selected information is “value”. In case of multihead attention multiple key value pairs are used and calculates plularity of information achieved from input in parallel and produc the final value. According to Vaswani et al., 2017, calculation for multi-head attention is as below

$$\begin{aligned} \text{MultiHead}\,(Q,K,V) &= \text{Concate}\,(\text{head}_1,\ldots,\text{head}_h)W^O, \\ \text{wherehead}_i &= \text{Attention}\left(QW_i^Q, KW_i^K, VW_i^V\right). \end{aligned} \tag{2.3.1-6}$$

- **Global Attention and Local Attention:** According to Luong et al., 2015, the idea of Global attention is to consider the hidden layer state of all encoders. The current decoder hidden layer state is compared with the state of each encoder layer from which it obtains the attention weight distribution. In the decoding process hard attention is similar to soft attention, which calculates the weight of the each word in the encoding and weights the context vector. The amount of calculation is comparatively large as it concentrates on all the encoder inputs when calculating every decoder state (Luong et al., 2015). In case of Local attention the alignment position is found and then the attention weight in the left and right window is calculated according to the position it is located and finally weights the context vector (Luong et al., 2015). The cost of the attention mechanism calculation is less, where it considers only the source language end which gets aligned in the current decoding based on the prediction function , it navigates through the context window considering the words within the window (Luong et al., 2015).

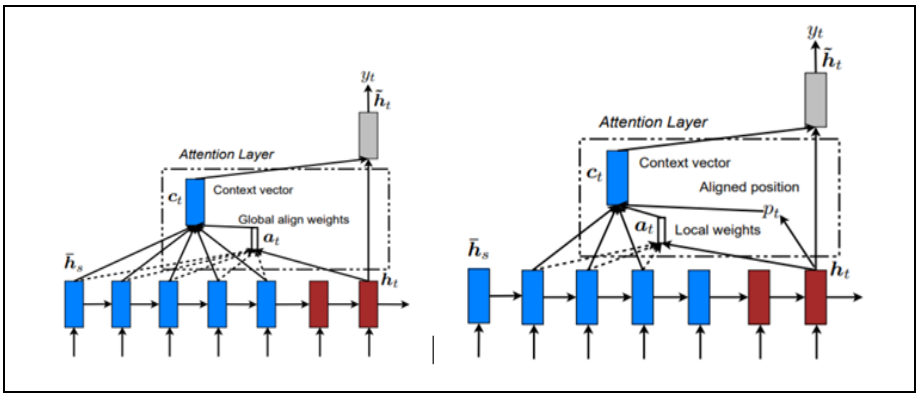


Figure 2.3.1 1 Global Attention (left) anf Local attention (right) taken from (Luong et al., 2015)

- **Spatial and Channel Wise attention :** Spatial and channel attention is the process of selecting semantic attributes according to the needs of sentence context (Chen et al., 2017). In order to overcome the general attention mechanism in decoding, attention mechanism is used according to the extracted semantics in the encoding process (Wang et al., 2020). Visual attention is obtained on multiple semantic abstractions as the mapping of the feature depends on the feature extraction and it applies attention on multiple layers.

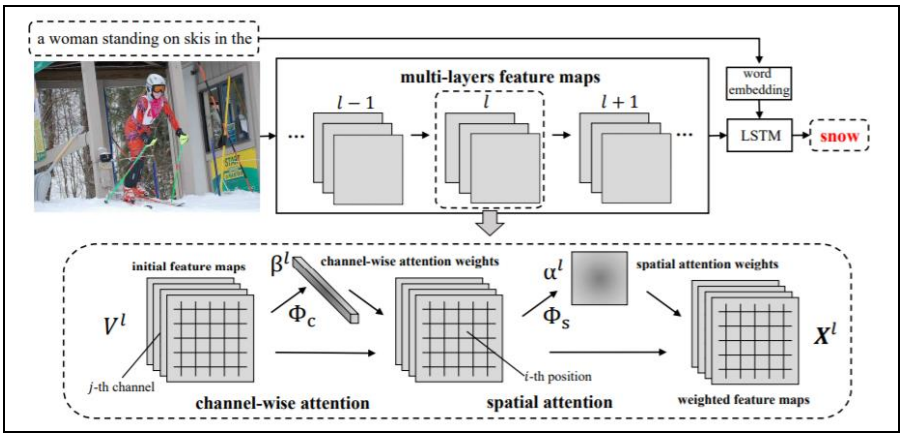


Figure 2.3.1 2 Spatial and channel wise attention CNN (Chen et al., 2017)

Text-Guided Attention: Mun et al. (2016), provide a new attention model for image captioning called text-guided attention model, which uses example captions in training data as a source of visual attention guidance. This exemplar-based attention model is taught end-to-end within the picture captioning system. Using several exemplar guiding captions, constructed a sampling-based technique for learning attention. This prevents overfitting in training and eliminates the problem of noisy guide captions

learning to mislead attention. In this architecture, there are two encoders: an image encoder and a guided caption encoder. For the picture and guiding caption encoders, model uses CNN and an RNN, respectively. A word embedding layer, an LSTM unit, and a word prediction layer make up the overall decoder. Assuming that the caption is made up of T words (w1, w2, ..., wT). s. the input word w_t is projected into the word vector space at each time step t. Based on the word vector x_t and the previous hidden state ht1, the LSTM unit calculates the current hidden state h_{t1}, then, based on the current hidden state h_t, the word is anticipated. In the following time step, the predicted word and the current hidden state are fed back into the decoder unit, and the process is repeated until the word is emitted. The guided captions highlight the key regions while suppressing unnecessary ones, allowing for precise caption generation. Throughout training and testing it offer a robust strategy for dealing with noise in guidance captions that uses a set of consensus captions as guide captions. On the MS-COCO Captioning dataset, this model achieved state-of-the-art performance with text guided attention model.

2.3.2 Transformer Based Image captioning

Transformers architecture is leading in natural language processing and have also gained its popularity in long-range representation and high performance. Instead of using RNN sequential architecture, transformers uses self-attention mechanism and are called sequence to sequence models (Xu et al., 2021).

- **Label Attention Transformer :** Dubey et al. (2021), proposed Label attention transformer with geometrically coherent objects for image captioning which establishes relationship between objects based on their localized ratios and encapsulates multi-level visual and geometrically coherent proposals. The objects are extracted from the image called proposal and these are assigned to known labels from classes which are passed through a label-attention module, finally for the detected proposals an effective geometrical relationship is computed.

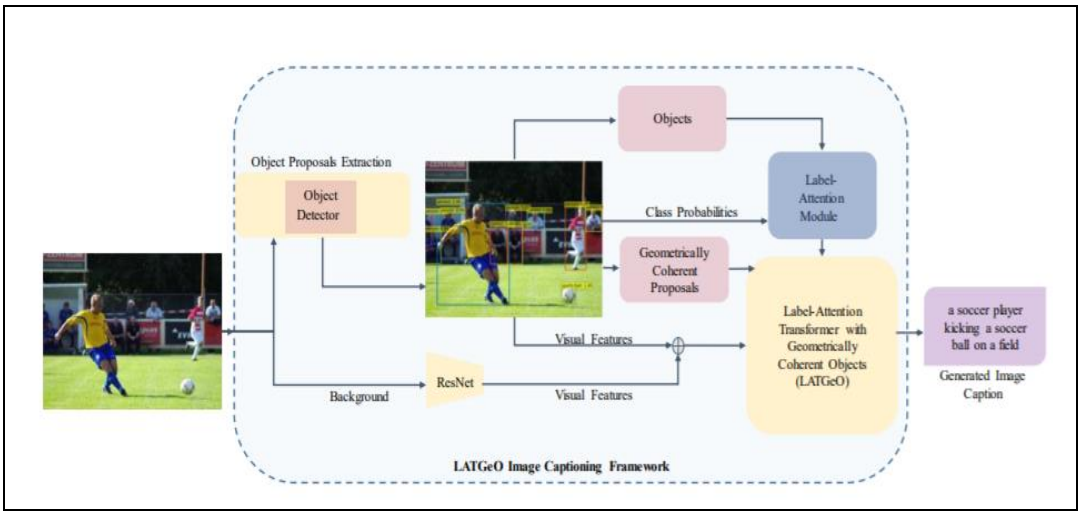


Figure 2.3.2 1 LATGeo architecture for image captioning Dubey et al., 2021

- **Geometry Attention Transformer (GAT) :** Wang et al. (2021) proposed an improved Geometry attention transformer with two novel geometry-aware architecture designed for encoder-decoder respectively. This model has two module one with geometry gate-controlled self-attention refiner which incorporates relative spatial information in to image region representation during encoding steps and other with group of position-LSTMs, informs the decoder of relative word position and generates caption texts.
- **Entangled Transformer:** Li et al. (2019) proposed entangled attention (ETA) transformer to exploit semantic and visual information. In this architecture attention is executed in an entangled manner and get effected by the preliminary modality when attention is performed over the target one. Variable number of semantic attributes are considered by selecting the multi-head attention as the preliminary information injection function. Semantic guidance performs multi head attention over the target modality.

$$\mathbf{g}_t^{(s)} = MultiHead(\mathbf{a}_t, \mathbf{S}^N, \mathbf{S}^N).$$

$$\mathbf{v}_t = MultiHead(\mathbf{g}_t^{(s)}, \mathbf{V}^N, \mathbf{V}^N), \quad (2.3.2\text{-}1)$$

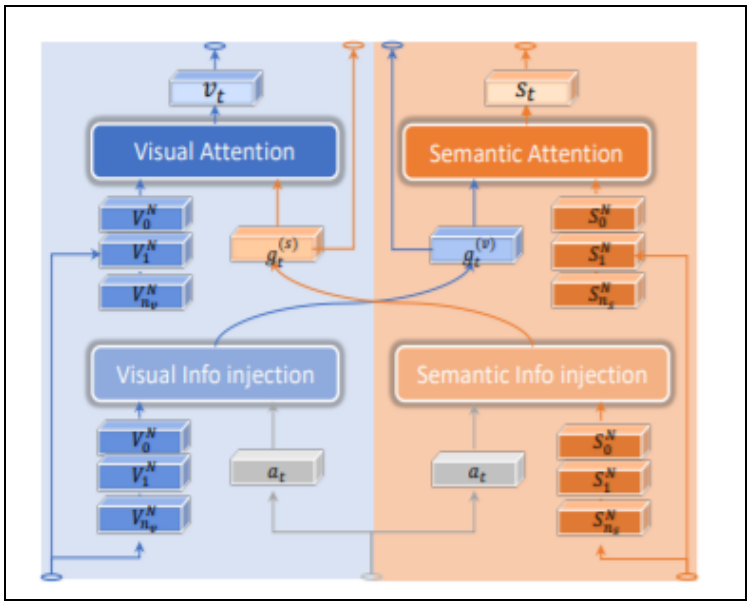


Figure 2.3.2 2 Entangled Transformer (Li et al., 2019)

- **CaPtion Transformer**: Liu et al. (2021) proposed CaPtion Transformer, where raw images are taken in sequentialized manner as an input to transformers. In comparison to the "CNN+Transformer" paradigm, CPTR is a more straightforward but effective technique that does not require any convolution. The CNN encoder has a constraint in global context modelling due to the local operator essence of convolution, which can only be overcome by extending receptive fields. As the convolution layers get deeper, the field becomes more complex. The CPTR encoder, on the other hand, can make use of long-range dependencies. In the cross attention layer of the decoder, CPTR models "words-to-patches" attention, which has been shown to be effective. In Encoder architecture rather than utilizing a pretrained CNN or Faster R-CNN model to extract spatial characteristics or bottom-up features as in earlier methods, author chose to input the image sequentially and treat image captioning as a sequence-to-sequence prediction challenge. Sinusoid positional embedding to the word embedding features in the decoder and uses both the addition results and the encoder output features as input. The decoder consist of identical layers (N_d) , each layer has masked multi-head self-attention sublayer , a multi-head cross attention sublayer, and a positional feedforward sublayer in that order (Liu et al., 2021). The last decoder layer's output feature is used to predict the

following word via a linear layer whose output dimension equals the vocabulary size.

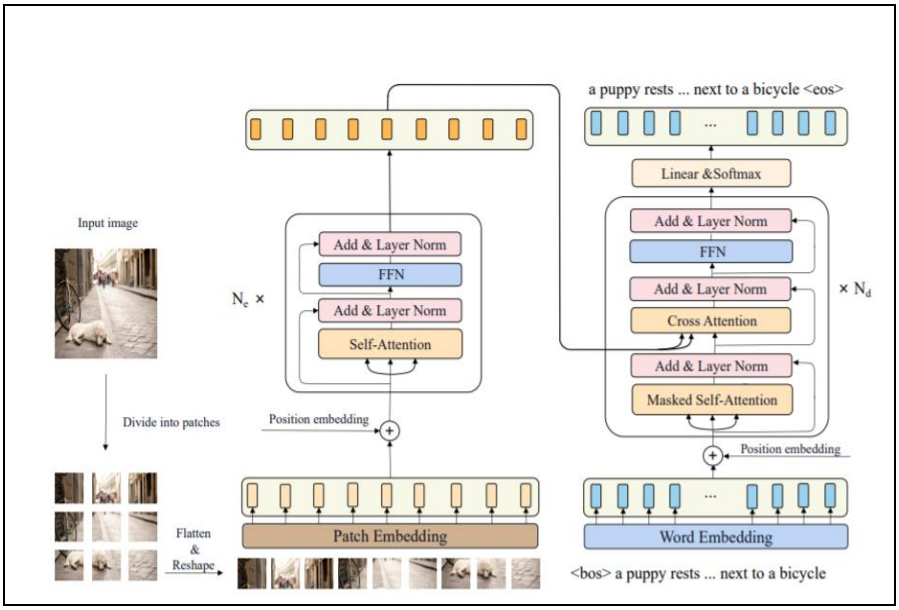


Figure 2.3.2 3 CPTR Architecture by Liu et al.

- **Meshed-Memory Transformer :** Cornia et al. (2020) introduced $\mathcal{M}^2$– a Meshed transformer with Memory for image captioning. Both the image encoding and language generation steps is improved by architecture by learning a multi-level representation of the relationships between image regions that incorporates learned a priori knowledge and exploiting low- and high-level features using mesh-like connectivity at the decoding stage (Cornia et al., 2020). In comparison to recurrent models, author analysed the performance of the $\mathcal{M}^2$ transformer and various fully-attentive models experimentally.

- **Sparse Transformer :** In this paper, a unique deep encoder-decoder model for image captioning is proposed by Lei et al. (2020), which is based on the sparse Transformer architecture. To exploit low-level and high-level features, the encoder uses a multi-level representation of image features based on self-attention. Naturally, the correlations between image region pairs are adequately modelled, as the self-attention operation can be seen as a way of encoding pairwise relationships. By explicitly picking the most relevant segments at each row of the attention matrix, the decoder increases multi-head self-attention concentration on the global context. It can assist the model in focusing on the

image regions that contribute the most and generating more accurate words in the context.

- **Multi-head Attention Transformer :** Vaswani et al. (2017) proposed simple network architecture solely based on attention mechanism based on transformer. The overall architecture of transformer follows using stacked self-attention and fully connected layers for both encoder-decoder. The encoder architecture is composed of a stack of six layers, where each layer has 2 sub-layers. The first layer has multi-head attention mechanism and second consists of fully connected feed-forward network. Each layer is then followed by normalization layer. Similar to encoder layer, decoder layer too has a stack of 6 layers and also has an additional sub-layer connected to the output of the encoder stack to perform multi-head attention. The attention function has scaled dot product and multi-head attention, which can be described for mapping a query and key-value pair to an output which are considered as vectors. The weighted sum of the output values are computed by assigning weight to each value by a compatibility function of the query with its corresponding key (Vaswani et al., 2017). The input of the scaled dot product attention consists of queries, keys and values of dimensions . The dot product of these are computed followed by the division of square root of the dimension of the keys. The weights on the values are obtained by applying the softmax function. Below is the attention function equation by Vaswani et al. (2017)

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \quad \text{(2.3.2-2)}$$

In case of multi-head attention mechanism projections of queries values and keys are made linearly with different learned projects of queries, key and values respectively and attention function is performed in parallel which yield dimensional output values. Below is the multi-head attention function equation by Vaswani et al. (2017)

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$
$$\text{where head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad \text{(2.3.2-3)}$$

Multi-head attention is used by the transformers in three different ways, the previous decoder layer in 'encoder-decoder attention layers' creates the queries, in this case all positions in the input sequence is attended by the decoder. All the keys, values and queries in the self-attention layer of encoder comes from same place, which in this case would be the output from the previous layer of the encoder, such that encoder would be able to attend all the positions from previous layer of the encoder (Vaswani et al., 2017). Similarly, decoder self-attention layer attends all each position of the decoder including that position. The decoder's left forward information must be prevented and needs to be reserved for the auto-regressive propoerty, which is implemented inside the scaled dot -product attention such that input values are masked out using softmax layer. Below of transformer architecture with multi-head attention mechanism (Vaswani et al., 2017).

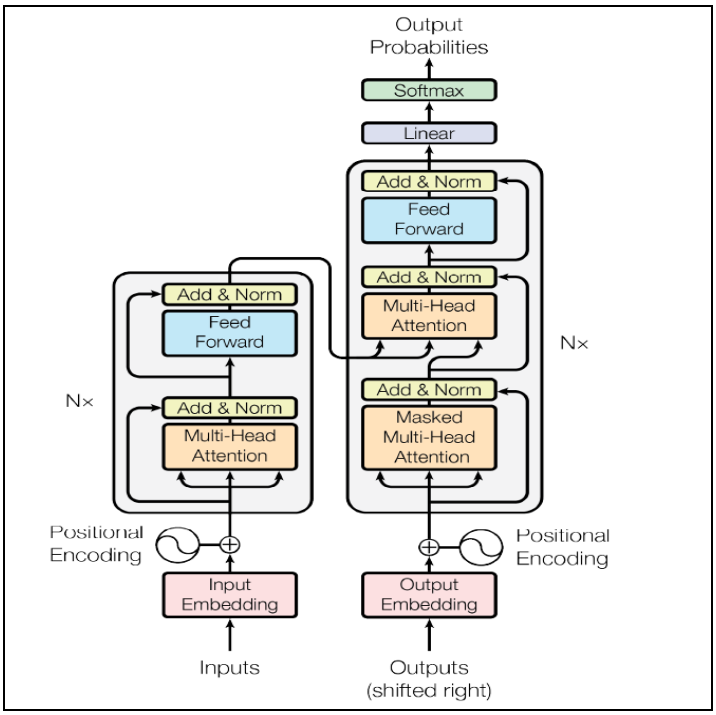


Figure 2.3.2 4 Transformer with multi-head attention mechanism taken from Vaswani et al., 2017

2.4 Evaluation for Image Captioning

BLEU, METEOR, CIDEr, ROUGE and SPICE are most commonly used evaluation metrics for sentence generation results. BLEU and METEOR are generally used for

machine translation, for automatic summary ROGUE is the most preferred metrics. For image captioning methods, BLEU, CIDEr and SPICE are used.

- **BLEU**: It is widely used evaluation metrics which was initially designed for the machine translation as it is based on accuracy rate evaluation. It find the correlation of n-gram between the reference and candidate statement (Papineni et al., 2001). The main idea of this metrics is, closer the statement to the human translation statement, it gives better score (Papineni et al., 2001; Wang et al., 2020).
- **METEOR**: METEOR is also used for machine translation, but unlike BLEU metrics it uses attention, recall and F values between the reference and candidate sentence. It considers the precision of n-gram and recall's harmonic mean, where recall's weight is bit higher than precision (Wang et al., 2020). Performance is better when METEOR score is high.
- **CIDEr**: CIDEr is widely used for image annotation problems. The consistency of the image captioning is measured with help of weight calculation of each -gram in Term Frequency-Inverse Document Frequency (TF-IDF). In this each sentence is considered as a 'document' and is represented in the form of TF-IDF and then the cosine similarity of the reference description is calculated (Vedantam et al., 2015).
- **SPICE**: SPICE is most widely used for image captioning evaluation. According to Anderson et al. (2016), all the above metrics are sensitive to n-gram, therefore proposes a new evaluation metrics called SPICE, which compares semantic propositional content and could perform better than BLEU, CIDEr and METEOR.

2.5 Summary

2.5.1Overview

There are many approaches proposed for image captioning, most of the paper in general focuses on simple encoder-decoder architecture which in general used LSTM as language model or CNN-RNN which unable to handle long sequence sentences. Novel captions can be generated from both multimodal and visual space. This

approach in general analyses the image content first and then using language model generate the image content form the visual content. These methods are able to generate new captions for each image with semantically accurate than previous approaches (Chu et al., 2020). In case of attention mechanism, distribution is based on probability distribution where it types varies according to how this attention needs to be calculated. For transformer model, inputs are sent in parallel and uses self-attention mechanism which can handle large sequence of data and perform better than other models.

2.5.2 Research Gaps

Few researchers have proposed template based approaches but these templates are fixed length and are predefined. Generating variable length captions is not possible with this method, however parsing -based language models can overcome the fixed length limitations (Hutchison et al., 2010;Li et al., 2011; Kulkarni et.al, 2013). Retrieval based methods were implemented in few of the papers, though these methods can generate syntactically correct captions, fail to generate captions specific to images and with correct semantics (Hodosh et al., 2013; Ordonez et al., 2011).

- Karpathy et al. (2014) has proposed bi-directional images and sentence retrieval which implements deep, multi modal embeddings of image and language data. It uses dependency tree relation (DTR) where images gets break down in to number of objects and sentences. Though this, model can handle relations easily but it cannot be precise and finding clear mapping in the image for many dependency relations would be difficult.

- Most of the research for image captioning are based on the basic encoder-decoder method which uses CNN for encoder and RNN is used for decoder in which word representation is converted in to natural language description of the model. Though this method is traditional approach and was successful to some extend but while describing the image it fails to analyse the image over time. These methods end up generating the caption as a whole without considering the spatial aspects of the image. Therefore, with the help of attention mechanism this limitation can be mitigated which focus on the various parts of the image while producing the output

sequence (Bengio et al., 2015; Kiros et al., 2015; Karpathy&Fei-Fei, 2015; (Hossain et al., 2019).

- Top-down and bottom-up approach are used by few researchers. In top-down approach visual features are extracted first and then converted in to words. In bottom-up approach visual concepts such as regions, objects and attributes are extracted from image and then combined. However, fine details are not captured by top-down approach which can be rectified by bottom up approach but in bottom up approach could not formulate end-end process (Anderson et al, 2018 ;Biswas et al., 2020 ; Zheng et al., 2019).

- Reinforcement learning techniques for image captioning are focused these days by researches which are designed with multiple parameters such as state, action, reward function, agent, value and policy. This method generally follows the method where agent choose an action and moves to next state based on reward value. Traditional reinforcement learning generally have limitations where there no guaranty for reward function and state-action information are uncertain (Shen et al., 2020;Xu et al.,2020;Yan et al., 2018). Advanced method of policy gradient reinforcement learning overcome such issue with gradient descent (Hossain et al., 2019).

- GAN (Generative Adversarial Network)-based methods are another novel methods which are used for various computer vision applications along with image captioning such as image to image translation, synthesis of text to image. However, using GAN methods has few issues, like GANs are known for real value data and processing of text is based on discrete numbers therefore the operation become nondifferentiable and making it difficult to use for backpropagation directly. Furthermore, it faces problem in vanishing gradient and sequence generation through error propagation (Wei et al., 2020; Yan et al., 2018).

- BLEU (Bilingual Evaluation Understudy) is the metric used for image captioning which measures the quality of text generated by deep learning models. Segments of texts are compared with a set of reference text and each of them are computed with the scores and are averaged. Vinyals et al., 2017 and Hossain et al., 2019 have used BLEU metrics for evaluation but this metric has few limitations, that BLEU score

can be used only when the texts are short and at some cases when the score is high does not mean that text generated is good. There are other few metrics which are far more advanced than BLEU and can overcome these limitations.

- **Human Evaluation :** Though there are many metrics to evaluate the model performance related to machine translation or image captioning, human evaluation would still be required for readability and quality of the output. Human evaluation can be of different type like intrinsic and extrinsic. In case of intrinsic evaluation , the real caption and the caption generated by the model is shown which is evaluated based on the ratings generated. In other type of intrinsic evaluation , captions generated by different models are presented and then asked to choose the best caption among them. Such evaluation could be carried out in form of questionnaire, survey or ranking systems to obtain the relevant caption generated by the models. In case of extrinsic evaluation the proposed model should be integrated with the real world, which can be time consuming and difficult to implement. Therefore, intrinsic evaluation is preferred than the extrinsic.

2.5.3 Research Question

From few of the researches and papers it is evident that attention model yields significant results for image captioning, therefore when transformer is combined with multi-head attention would further improve the model.

“To what extend BLEU (Bilingual Evaluation Under Study) score for image captioning with deep learning can be improved by augmenting the input data from image captioning dataset like Flickr8K using transformers with multi-head attention model when compared to attention model?”

3. DESIGN AND METHODOLOGY

In this chapter will be describing the experiment used to determine whether the null hypothesis can be accepted or rejected. For the purpose of image captioning will experimenting with three different methods one is using a simple RNN-CNN architecture, next will be attention mechanism and in third will be using transformers. Data collection and preparation for conducting this experiment is described followed by the detailed explanation of the models for the purpose of the experiment.

Finally will be evaluating the results obtained from the attention mechanism and Transformers using BLEU score metrics. For this purpose will be conducting statistical t-test for hypothesis testing, followed by human evaluation with the help of a survey to understand how well the model has captioned the image and how does human rate the predicted caption.

3.1 Hypothesis

Null Hypothesis(H0):If a transformer with multi-head attention model is used to augment the input data based on image captioning dataset from Flickr8K, t-test obtained from the BLEU score is not statistically significantly higher than the t-test obtained from the BLEU scores associated to image captioning without such augmentation like attention model

Alternate Hypothesis (H1): If a transformer with multi-head attention model is used to augment the input data based on image captioning dataset from Flickr8K, t-test obtained from the BLEU score is statistically significantly higher than the t-test obtained from the BLEU scores associated to image captioning without such augmentation like attention model

3.2 Data Collection, Understanding and Preparation

Numerous datasets are used for training, testing and evaluation of the image captions. These datasets may vary in various perspectives such as total number of images in the datasets, number of captions per image, size of the image. The most commonly used datasets are Flickr8K, flickr30K and MSCOCO dataset. For this experiment, will be

using Flickr8k dataset as this will be idle for the training, testing and obtaining the scores within the limited time period. Flickr8k dataset has 8000 images, which has image id with 5 captions each. This dataset is a good start for the beginning of the image captioning and is also relatively small compared to other datasets. During data preparation will be creating a dataframe to store the images and captions so that captions for the images could be seen together. The dataset for this experiment is taken from the website[2] which can be downloaded in zip format.

Visualizing few images present in the dataset.

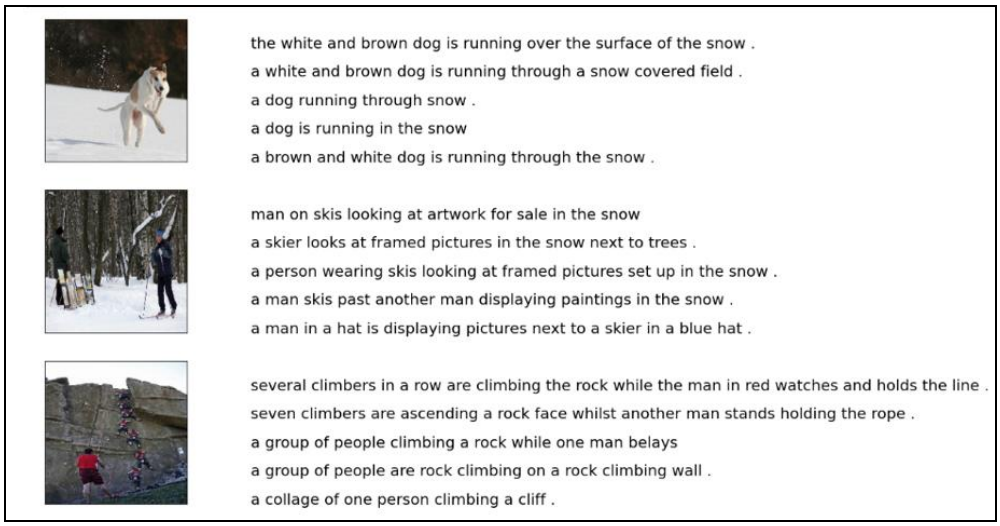


Figure 3.2 1 Sample Images and its captions present in Flickr8k dataset

Vocabulary size of the entire set is 8918 without cleaning the dataset. Then need to apply clean function to the captions in the dataset to remove any punctuations, characters and numbers for the efficient performance of the model. After cleaning the dataset the cleaned vocabulary size is 8357, more than 400 words has been eliminated. <start> and <end> sequences are appended to each captions for model to understand the start and end of the sentences.In Flickr8k dataset there are totally 8000 images with 5 captions each, will be combining all together to make in 40, 000 images with 40,000 captions and will be storing a variable for creating a train and test dataset with 80:20 split.

[2] https://machinelearningmastery.com/prepare-photo-caption-dataset-training-deep-learning-model/

3.3 CNN-LSTM Model

Neural Network architecture consists of series of convolution layers which has layers of nonlinear and pooling layers, where image gets passed through the one convolution layers and its output becomes the input for the second layer. The fully connected layer needs to be attached after the series of convolutional, nonlinear and pooling layers. The information gets passed through the fully connected layer and results in an N dimensional vector, where N refers to the number of classes.

LSTM (Long Short term memory), is a type of RNN, which has a ability to learn the order sequences in case of sequence predictions. LSTM architecture is preferred here over traditional RNN because LSTM overcomes the short term memory limitations and when we go deeper into a neural network, gradients would be small or zero where chances of training will be less and would lead to poor predictive performance. LSTM have the ability to discard irrelevant information with the help of forget gate and would be able to carry relevant information throughout the processing of inputs.

- CNN-LSTM Architecture: CNN-LSTM architecture is generally used when inputs have spatial structure such pixels in the image or 2D structure or words in paragraph or sentences which has 1D structure and should also have a temporal structure like order of images in a video or words in text. In this approach will be using the concept of CNN and LSTM and will build an image captioning generator with the help of natural language processing which will understand the context of the image and would be able to describe it in English and will use the concept of computer vision.

Image captioning task is generally divided in two modules such as image based module where the feature of the image will be extracted, next is language based model in which extracted feature and objects gets converted in to natural sentence. In CNN-LSTM architecture CNN is used for image based model and LSTM is used for language based model.

CNN is used as an encoder which extracts the feature if the image and RNN is a decoder is used for the image description. Here we have used inception V3 which is a pretrained vector and no need to create a layer by ourselves. The CNN output is fed to RNN that learns to generate words. Given the image, every vertical layer tries to predict the next word. The first layer will take the embedded image and will predict the

start word, which is followed by other tags. In order to have a long term memory LSTM uses cells are been used which helps to keep the state of the word.

In order to train our LSTM model, the target text and label gets predefined. If the caption is “a girl going in to the wooden house”, then the label and the target would be the following:

Label-[<startseq>,A, girl, going, in, to, the, wooden, house.]

Target-[A girl going in to the wooden house.,<endseq>]

. Will be following the below methods for the image captioning:

- Image Preprocessing
- Generating vocabulary size
- Model Building and Training
- Model Evaluation
- Image captioning on Individual Images

This helps the model to understand the start and end of the labelled sequence. The dataset used here is Flickr8K dataset. Inception V3 is used as pretrained model, which is already installed in the keras library. Images features are in size 224*224 size, where features are extracted before the classification layer because this pretrained model is used for classification of an image therefore will be excluding the last layer as we are not interested in classification of the image.

- Model Building and Training: For this approach we are building CNN-LSTM model which will predict the sequence of words or captions from the feature vectors obtained from the inception V3. The below figure shows CNN-LSTM architecture taken from (Vinyals et al., 2017) , where deep convolution network is used to creat a semantic representation on the image and is decoded using LSTM network.

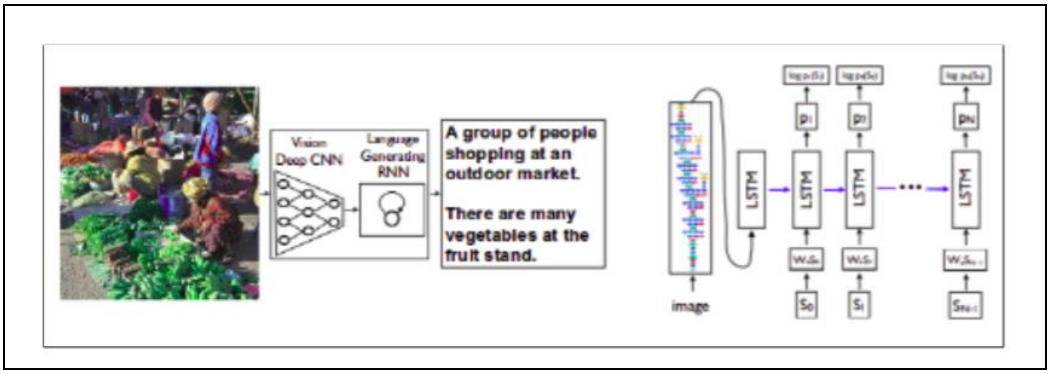


Figure 3.3 1 Deep CNN-LSTM architecture taken from Vinyals et al., 2017

In order to train the model, will be using 6000 training images in 3 batches and will fit the model and train the model with 30 epochs. Below the model summary.

```
model.summary()

Model: "model_1"
```

Layer (type)	Output Shape	Param #	Connected to
input_3 (InputLayer)	[(None, 38)]	0	
input_2 (InputLayer)	[(None, 2048)]	0	
embedding (Embedding)	(None, 38, 200)	379600	input_3[0][0]
dropout (Dropout)	(None, 2048)	0	input_2[0][0]
dropout_1 (Dropout)	(None, 38, 200)	0	embedding[0][0]
dense (Dense)	(None, 256)	524544	dropout[0][0]
lstm (LSTM)	(None, 256)	467968	dropout_1[0][0]
add (Add)	(None, 256)	0	dense[0][0] lstm[0][0]
dense_1 (Dense)	(None, 256)	65792	add[0][0]
dense_2 (Dense)	(None, 1898)	487786	dense_1[0][0]

```
Total params: 1,925,690
Trainable params: 1,925,690
Non-trainable params: 0
```

Figure 3.3 2 Model Summary of CNN-LSTM

Below are the few image captions generated with simple CNN-LSTM architecture.

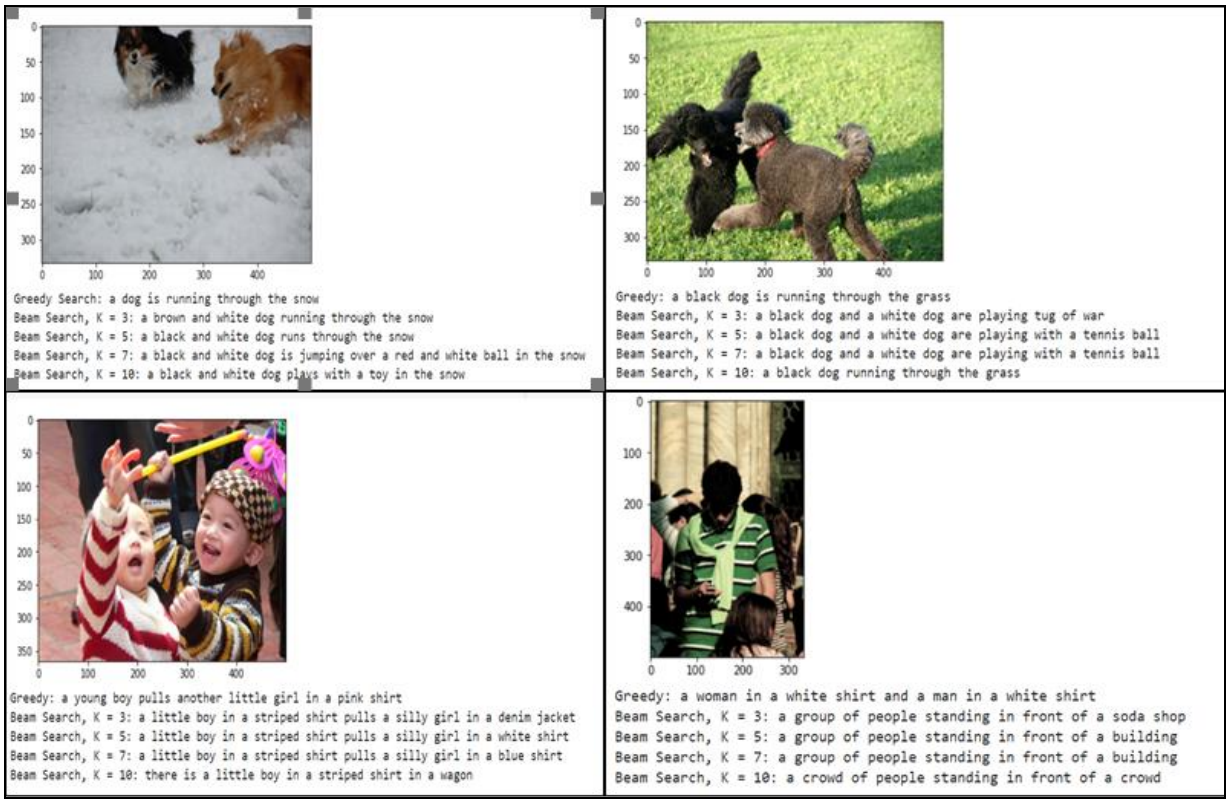


Figure 3.3 3 Image Captioned by CNN-LSTM model using Greedy and Beam search

3.4 Attention Mechanism

In case of encoder decoder architecture, decoder uses the hidden state from previous time step and produces output word for current timestep. As per the website mentioned below in the image the representation of the encoded image feature is carried by the hidden state. All image gets treated similarly when the out word is created in the decoder, but with the help of attention module encoded image is taken at each timestep as input with the hidden state from prior timestep in the decoder. During this stage, score from the attention is produced which assigns a weight to each pixel from the encoded image, higher the weight to the pixel signifies the relevancy of the word as the output at the next timestep.

For example if the target output sequence is 'boy in a blue dress', then the boy's pixels will be highlighted for the word "boy" and same for the blue dress. Then the score gets concatenated along with the input word of that timestep, then fed to decoder. Decoder generates the appropriate output word by directing on the most appropriate parts of the image.

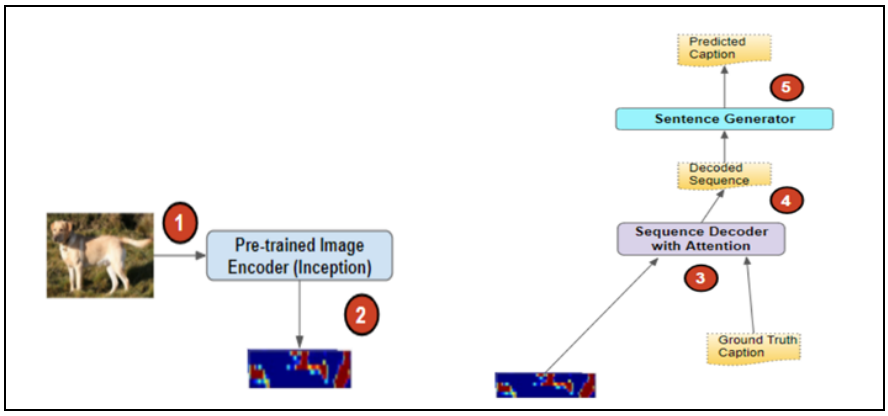


Figure 3.4 1 Simple attention architecture taken from a website[3]

In this approach components similar to simple encoder decoder will be but still be including the extra component called attention mechanism. During this method, image files from Flickr8K dataset will be given as input and their essential features will be enhanced further with the help of attention mechanism. Similar to first approach, transfer learning method will be used for pre-processing the raw images with the pre-trained network using CNN therefore the network does not need further training. Few of the pretrained models that are available are VGG16,VGG19, ResNet, InceptionV3

[3] https://towardsdatascience.com/image-captions-with-attention-in-tensorflow-step-by-step-927dad3569fa

etc. For this experiment will be implementing VGG16 model as a pretrained model, which is convolution neural network and achieved 92.7% accuracy in ImageNet that is a dataset of 14 million images with 1000 classes, it gets trained for weeks. In VGG16, input layer is fixed size with 224*224 RGB image, the image gets passed through the stack of convolution layers and contains softmax layer at the end.

Bahdanau et al., 2016, has proposed the idea of attention mechanism with fixed length vector unlike traditional approaches with variable length vector where the information from the source gets lost. In this approach, instead of encoding the whole sentence in to one single length vector, input sequence gets encoded in sequence of vectors which is the most distinguishing feature of this approach. There are few major components in Bahnadau encoder-decoder architecture, these are:

- hidden decoder state s_{t-1} at previous time step (t-1).
- At time step t, there is a context vector c_t, that gets generated uniquely at each decoder step and generates the target word y_t.
- Annotation h_i, is useful in capturing the important information from the input sentence containing the words which focuses on the i^{th} word out of total words.
- Weight values $\alpha_{t,i}$ gets assigned to each annotation h_i, at time step t
- Attention score $e_{t,i}$ is used to evaluate on how well s_{t-1} and h_i match

The above components gets used in the Bahnadau's architecture having bidirectional RNN as encoder and an RNN as decoder and attention mechanism in between them. The below architecture is taken from Bahdanau et al., 2016, depicting the use of above mentioned components.

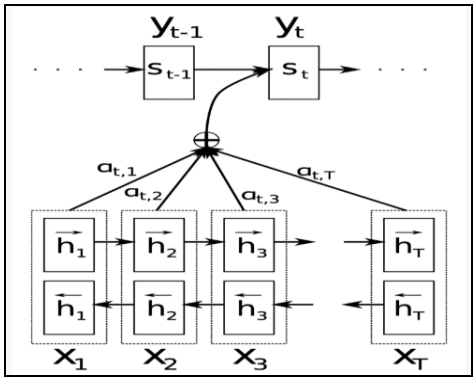


Figure 3.4 2 Bahdanau et al., 2016 Attention architecture

In summary, a set of annotations h_i are generated by encoder from input sentence, then these annotations along with the previous hidden decoder state are fed to alignment model together to generate the attention scores, $e_{t,i}$. Attention scores gets normalized effectively in to weight values $\alpha_{t,i}$ by softmax function to range between 0 and 1. The context vector c_t is generated through weighted sum of annotations. Then the final output y_t is generated when context vector is fed to the decoder along the previous hidden decoder state and the previous output. These steps gets repeated until the end of the sequence.

For the image caption model, training data consists of encoded feature vectors (x) and the captions which are the targets (y). During training, the images and captions will be loaded and pre-processed. The pre-processing steps is similar to the encoder-decoder architecture, only during model building attention mechanism will be used for further enhancement.

3.4.1Hyperparameters and Training-Attention Architecture

The functions based on the architecture mentioned by Bahdanau's attention mechanism needs to be defined. VGG16 encoder function needs to be defined first, then need to define RNN based GPU/CPU capabilities and then the RNN decoder. In RNN decoder, hidden state and the decoder input which is start token are passed. Parameters used for attention model are:

- Batch Size = 64
- Buffer size=1000
- embedding-dimension = 256
- Units=512
- feature shape=512
- attention feature shape = 49.

Hence, the output from the encoder is in shape (64,49,256), the hidden shape (batch size, hidden size) will be in shape (64,512),attention score will have a shape (64,49,1), then will be applying softmax function to the attention score for it to range in between 0 to 1 and dropout layer is added to avoid overfitting of the model. Categorical cross entropy is used as loss function

During training, will be creating a training loop which will train the model. Optimizer and loss functions will be defined. Model will be trained in 30 epochs and batch processing will happen in each iteration.

Data elements required for this will be setup first.

- Will be using the encoder decoder architecture at initial stage, next each element of the input sequence will be iterated through each element over multiple timesteps.
- Attention module computes the attention score with help of encoded image from the encoder and the sequence decoder from the hidden state. The input sequence passes through embedding layer and is then combined with the attention score
- Sequence decoder gets the combined input sequence and then the output sequence is generated along with the new hidden state. This output sequence is then processed through the sentence generator and generates the predicted word probabilities.
- This cycle is repeated until ‘endseq’ token is predicted or maximum length of the sequence is reached.
- The predicted probabilities are compared with actual captions of the image to calculate the loss, this will be used with back propagation to train the network.

3.4.2 Evaluation of Attention Model

BLEU (Bilingual Evaluation Under study) metric is implemented for evaluating the performance of the model, which is efficient in evaluating a generated text to a reference sentence. When the score is near to 1 then it indicates that model has predicted well. There are different methods to calculate BLEU score like Individual- N grams and cumulative-N grams score. In individual N-gram score, matching grams of specific order is evaluated and in cumulative individual n-grams are scored at all orders from 1 to n with weighted geometric mean.

For this experiment have implemented cumulative n-gram score. Below are the few examples of the image generated by the attention model with BLEU score

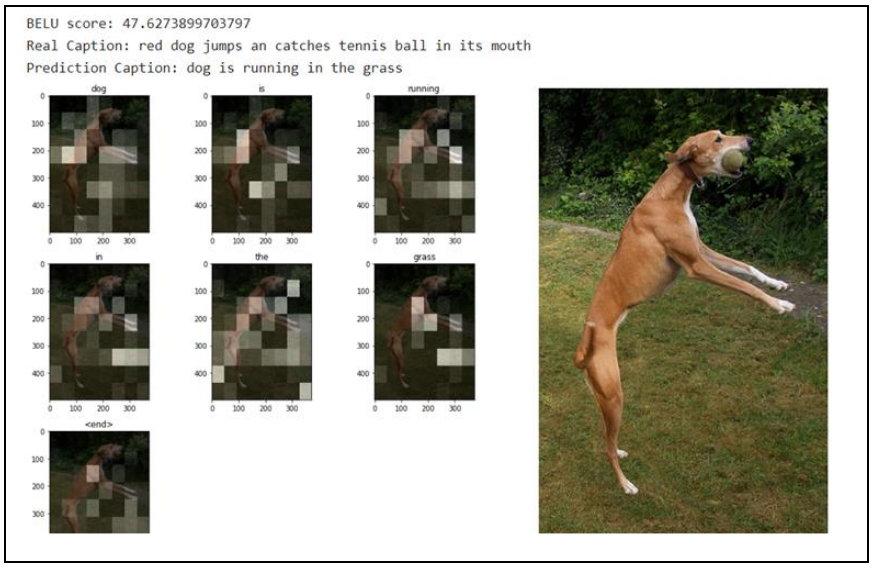


Figure 3.4.2 1 Image caption generated by Attention Model with BLEU score

The above image have BLEU score of 0.47, which is not a perfect match but still with the attention mechanism, model is able to predict dog and grass correctly and has predicted that 'dog as running' instead of jumping and could not identify ball in its mouth. Looking into few more examples of images captioned by attention model.

Figure 3.4.2 2 Image caption generated by Attention Model with BLEU score

From above few examples, it shows that model has performed pretty well and has captioned the images which is approximately matching with the real caption. In Figure 3.4.1-1(left), have BLEU score of 79% (0.79), shows that model is able to identify surfer, beach and waves. In case of Figure 3.4.1-1(Right), have BLEU score of 62% (0.62),indicates that model could able to identify that 'man is riding' but did not recognize gondola, instead it captioned tall buildings nearby which is not present in the real caption, which resulted in in less BLEU score. Therefore, in case of BLEU metric score it also depends on how the real captions has been identified. In few cases BLEU

score might be very low but still the image caption generated by model would be relevant and acceptable.

3.5 Transformers Architecture

Transformer architecture excels in handing the text data and is inherently sequential. Transformer network contains stack of encoder-decoder architecture similar to RNN. This group of encoder and decoder have their embedding layer for their respective inputs, finally the output layer generates the final output.

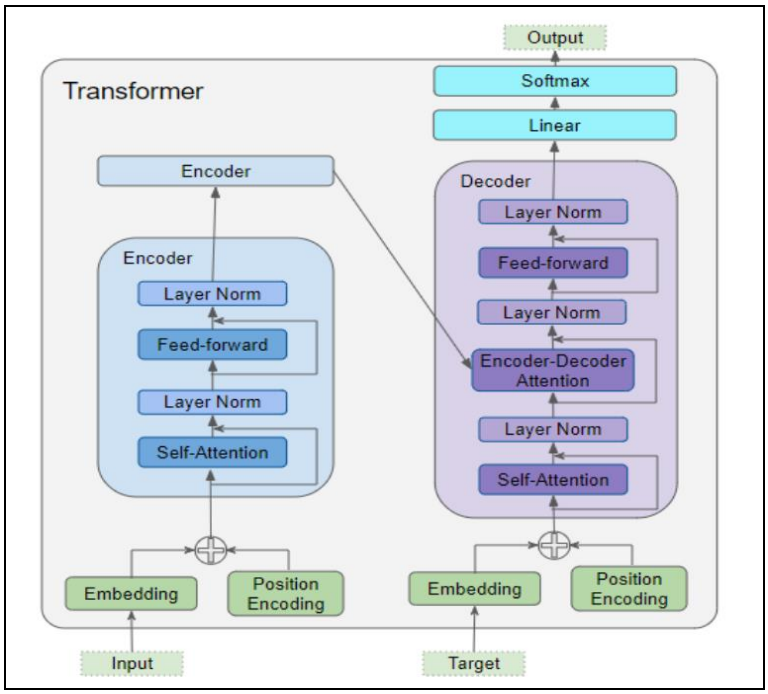


Figure 3.5 1 General Transformer architecture taken from website[4]

Data inputs for both encoder and decoder contains embedding and position encoding layer. Encoder stack containing number of encoder layer contains multi-head attention layer and feed-forward layer, similar to encoder decoder stack contains two multi-head attention and feed forward layer. The output layer generates the final output contains linear layer and softmax layer. The embedding layer in the transformers encodes the meaning of the word and position of the word is taken care by position encoding. The sequence of text is mapped to numeric word ids using vocabulary, embedding layer helps in mapping the word in to embedding vector. RNN generates a

[4] https://towardsdatascience.com/transformers-explained-visually-part-2-how-it-works-step-by-step-b49fa4a64f34

loop where each word is input sequentially and so it knows the position of each word. However, in transformers RNN is not used, all the words are input in parallel instead of sequence, which causes it to lose the information of the position of the word. There are two position encodings layers and are computed independently of the input sequence. The constants values are computed using below formula, taken from the website mentioned in the above figure, where 'pos' refers to word position, 'd_model' is the encoding vector and 'I' is the index value into the vector.

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}}) \quad (3.5\text{-}1)$$

- **Encoder**: There are stacks of encoder and decoder layers which are connected sequentially. The encoder in the first stack which receives the input from the position encoding and embedding, the output from this encoder is the input to the next encoder. Encoder has a multi-head self attention layer, feed forward layer and the normalization layer. The output from the last encoder is fed in to each decoder in decoder stack

- **Decoder**: Decoder structure is very analogous to encoder structure. The input to the first decoder in the stack comes from embedding and position encoding. The output of this decoder is then used as the key into the stack's next decoder. The decoder's input is transmitted through the multi-head self-attention system, which acts in a slightly different way. It can attend only the earlier position in the sequence and is achieved by masking the future positions. Decoder structure have a second multi-head attention layer which is called encoder-decoder attention layer. It receives two sets of input one is the output from the encoder stack and the self-attention layer below the encoder-decoder attention layer. A residual skip-connection layer is present in all the layers, followed by a normalization layer.

- **Attention**: In transformers, use of attention is the key to its ground breaking performance. During processing of the word, attention helps the model to focus on the other input words that are closely related. The self-attention in the

transformer architecture helps in relating every word from the input sequence to the other words.

For example:

- The dog ate the food because it was hungry
- The dog ate the food because it was delicious.

From the above two sentences one can notice that in the first sentence ‘it’ refers to dog and in second sentence it refers to food. Therefore while processing the word, self-attention gives more information to the model about the meaning of the sentence so that it can be associated to the correct word. In order to handle such nuances multiple-attention score is used by transformers for every word.

The query, key, and value parameters provide input to the attention layer, with encoder self-attention having its input passed through all three parameters, and decoder self-attention having its input passed through all three parameters. In the case of a decoder with an encoder-decoder layer, the output from the final encoder stack is passed through the key and value parameters, while the output from the decoder's self-attention layer is passed through the query parameter.

- **Multi-head attention:** Attention processor gets used several times in a Multi-head attention transformer. The linear layer gets the input as query, key and value with their own weights. These are combined together to form the attention score. The encoded representation of each word in the sentence is carried by the parameters that are passed through linear layer which are query, key and value. During the calculation of the attention score, masking is applied just before the softmax function, such that masked out elements set to negative infinity and turns to zero using softmax function.

- **Output Generation:** Output generated from the decoder stack is passed to the output component and is converted to final output sequence. The decoder vector gets projected to word score by linear layer and thus a score value is created for each unique word in the target vocabulary, according to each position in the sentence. For example if target vocabulary has some unique words (8000) and the final output has 8-10 words then that score values for each of the those words are created. The score values are the chance of

occurrence of each word in the vocabulary in a particular position of the sentence. These score are then converted in to probabilities by softmax layer. The index of the word with highest probability in each position is identified and gets mapped to the corresponding word in the vocabulary to form the sequence output from the transformer.

3.5.1 Model Definition, Hyperparameter and Training for Transformer Model:

Transformer follows different pattern during training, where data flows in two parts one is input sequence or source and other is output sequence or target sequence.

The data gets processed in the transformer using position encoding that converts the input sequence in to embeddings. The positional encoding makes use of sine and cosine functions with different frequencies. Every odd index in input vector gets created using cos function and even index in the input vector gets created using sin function. Then the network information on the position of each vector is fetched by combining the vectors to their corresponding input embeddings. The input sequence which is converted in to embeddings is fed in to set of encoder and the encoded representation of the input sequence is produced. Then the target sequence combines with the ‘startseq’ tokens and gets converted in to embeddings with positional encoding and fed in to the decoder, where decoder along with the encoder produces encoded representation of the target sequence. The output layer help in converting the target sequence to word probabilities and the final output sequence. The loss function of the transformer compares target and the output sequence, therefore gradients are generated using the loss and transformer gets trained during back propagation.

- **Model Definition:** Image feature extraction model can be done using a pretrained model using inception v3, which is efficient model for transfer learning and can be used for this experiment. For image captioning, only the image vector needs to be extracted and do not to classify it, therefore softmax layer from this model can be eliminated. The images needs to be pre-processed to same size i.e. 299*299 before feeding to the model. Output shape from this layer will be 8*8*2048. As mentioned in the above section , all the captions needs to be tokenized to build a vocabulary having a unique words. Train and

test data are split in 80:20 ratio. Need to define a position encoding function of different frequencies. Every odd index from the input vector creates a vector using cos function and sin function is used for every even index. Then need to create multi-head attention function with attention weights q, k ,v which should have a matching dimensions. Then the encoder-decoder and transformer function is defined.

- **Hyperparameters:** number of layers used is 4, d_model (encoding vector) is 512. Total number of features 2048 and 8 attention heads are used. Optimizer used for the model is Adam optimizer, with categorical cross entropy function and dropout layer is added to avoid overfitting of the model

- **Model Training:** Model is trained with 30 epochs, which took 4 hours in colab notebook with GPU runtime settings.

3.5.2 Transformer Model Evaluation

Similar to attention model, BLEU metrics will be used to evaluate the model performance.

Few of the outputs using transformer model along with BLEU score.

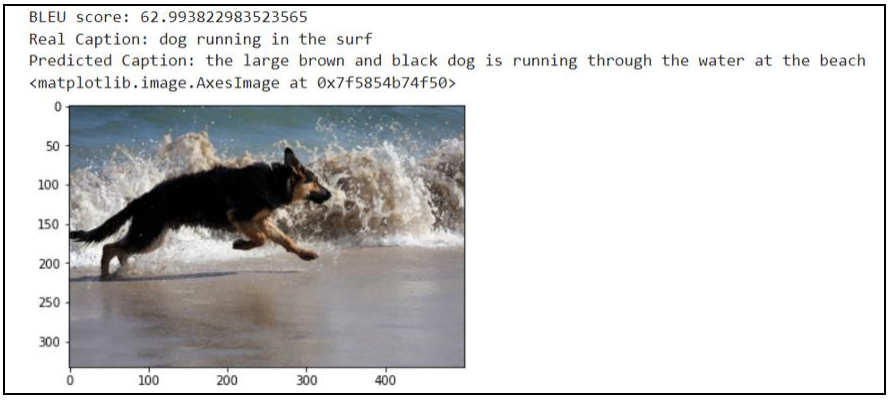


Figure 3.5.2 1 Image caption generated by Transformer model with BLEU score

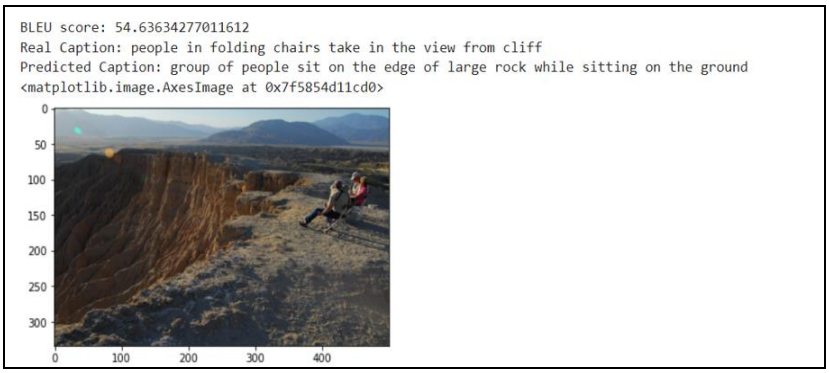


Figure 3.5.2 2 Image caption generated by Transformer model with BLEU score

The output generated from Transformer is much efficient , as it can be observed that in Figure 3.5.2-1, the score is 0.62, the predicted the image caption is more appropriate than the real caption. The model could able to differential the colour of the dog as black and brown, it could even able to identify that the dog is running along the beach, which gives the accurate caption for the image. In case of Figure 3.5.2-2 the score is less compared to the first one, as the model could not able to identify the cliff, instead it has identified the cliff as large rock and did not able to identify the chairs. Otherwise, model has identified that group of people are sitting at the edge.

From the above results, it shows that Transformer architecture could able to pay more attention to details in the image compared to attention model. In the coming section , will be comparing the overall BLEU-1, BLEU-2, BLEU-3, BLEU-4 mean score of attention and transformer model.

3.6 Summary

In order to conduct the experiment, the Flickr8k dataset has been gathered and is pre-processed in to embedding vectors. Then the simple CNN-LSTM model, attention model and transformer model is defined with suitable model parameters and trained with 30 epochs. Once completing the model training greedy and beam-search method is implemented to for the CNN-LSTM model for image captioning. In case of attention and transformer model, BLEU metrics score and human evaluation are used for analysing the performance of the model.

3.6.1 Strengths

Few strengths of simple CNN-LSTM model is it is easy to implement and train. In case of attention model it gives the model in focusing on the relevant part of the image instead of focusing on the entire image and its features. The implementation of local attention reduces the cost of attention mechanism calculation, as during the calculation, all the words in the sentence are not considered, only the position of the word in that window is focused which makes it fast and easy to implement. In case of transformer model, instead of receiving the input in sequence it is received in parallel therefore there is no time step associated and makes it computationally more powerful than the other two models.

3.6.2 Weakness

Simple CNN-LSTM model for image captioning fail to attend the longer sentences and it starts to repeat the words while captioning the image, In case of attention of model, when the input sentence is long then it add more weight parameters to the model which increases the training time. In case of transformers, it has limited access to high level representations. Since, all the inputs are attended in parallel, layer by layer as part of encoder and decoder stack transformer does not have the leverage to get the highest level of representation from the past and to compute the current representation (Rush et al., 2015). Sometimes, in case of transformers, bias in the tokenization can occur as tokenization is done for the whole set of the captions to avoid any unknown tokens.

4 RESULTS, EVALUATION AND DISCUSSION

Process of experiment will be examined in this section and will be compared for the hypothesis testing. For the hypothesis testing will be using t-test between the BLEU scores obtained from the Attention model and transformers model. Furthermore, will be discussing the strengths and weakness based on the findings while conducting the experiment.

4.1 Results

In the above sections have discussed about how the image captions generated by attention model and transformer model. In this section will be comparing and discussing the few of the captions generated by the model for the same set of images.

Figure 4.1 1 Image Caption generated by Attention model and Transformer model

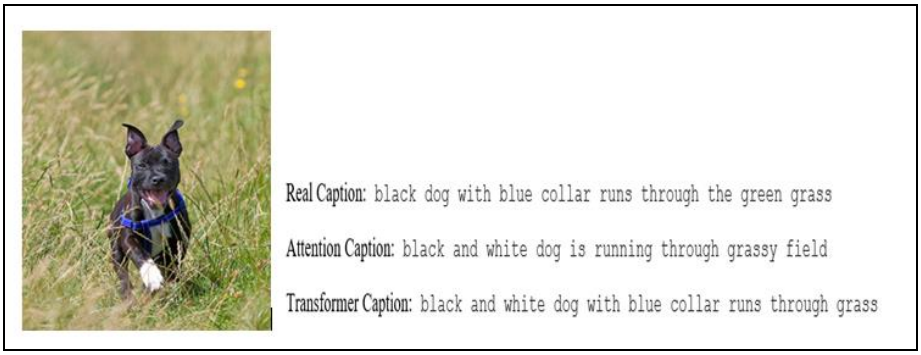


Figure 4.1 2 Image Caption generated by Attention model and Transformer model

In the above Figure 4.1-1 it can be noticed that Transformer model has captioned the image accurately and even better than the real caption., for the same image attention model has identified the dog as running instead of jumping. In case of Figure 4.1-2, both attention and transformer has identified the color of the dog as black and white,

but attention model has failed to recognize the blue collar but Transformer has captioned it perfectly.

Now comparing the images with BLEU score for attention model and Transformer model.

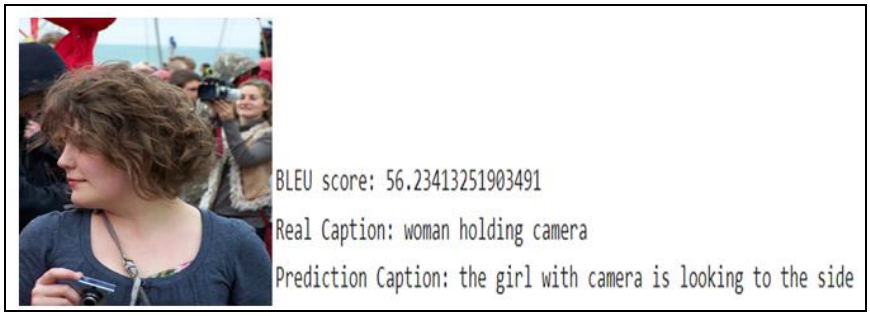


Figure 4.1 3 Image Caption by Attention model with BLEU score

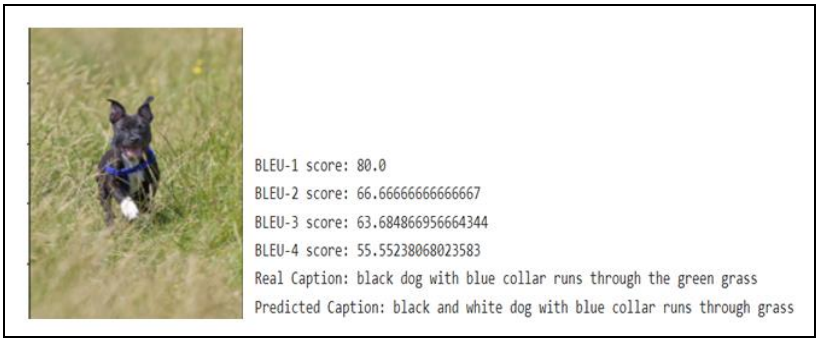


Figure 4.1 4 Image caption by Transformer model with BLEU scores(N=1,2,3,4)

In Figure 4.1-3, the real caption is very small, however the caption generated by the Attention model have more details and is accurate. However, the BLEU score is still not at its best, which is the down fall of the BLEU score metrics as it looks for the exact words and matches the n-gram and updates the scores based on precision. In Figure 4.1-4, BLEU-1 score is very high because BLEU metrics compares each word (1-gram) in the predicted caption with the real caption and looks for the exact same words. In case of others, score is comparatively less because the BLEU score starts to compare in pairs. However, the caption generated by the Transformer model is more appropriate as it could able to identify the colour of the dog as black and white, which was failed to notice even in the real caption. Therefore, from the above two examples it can be argued that just the BLEU metrics won't be enough to come to a conclusion. Analysing the results, using Human evaluation would be required for a transparent evaluation.

BLEU score for each of the predicted captions are obtained against the real captions and mean score is calculated. Below are the mean BLEU scores generated by each model

Model\BLEU N-Gram mean score	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Attention	0.199	0.257	0.357	0.393
Transformer	0.230	0.292	0.385	**0.418**

Table 4.1 1 BLEU mean score between Attention and Transformer

4.2 Evaluation

From the Table 4.1-1, it shows that Transformer has outperformed Attention model in image captioning with mean BLEU score of 0.418. But in order to compare the score statistically , will be conducting the t-test to accept or reject the null hypothesis based on the p value obtained from the t-test. BLEU score for the test set of 8000 images were obtained using attention model and transformer model trained with 30 epochs. Then these score were compared using t-test for hypothesis testing to determine if there is a significant difference between the means of two groups.

The p-value obtained from the t-test is 0.007 which is less than 0.05 (significance level) and will be rejecting the Null hypothesis. The t-test conducted between the bleu score obtained from the attention and transformer model proves that “if a transformer with multi-head attention model is used to augment the input data based on image captioning dataset from Flickr8K, t-test obtained from the BLEU score is statistically significantly higher than the t-test obtained from the BLEU scores associated to image captioning without such augmentation like attention model”.

4.3 Human Evaluation

Human evaluation has been done during the experiment, to determine which model has performed well. For this, a survey website has been used and images along the captions are generated by attention model and transformer model are presented to few people (approximately 10), where they can choose the best caption from the option

provided, by looking in to the image. Once completing the survey, the results were obtained and evaluated.

Below are the example results obtained from the survey

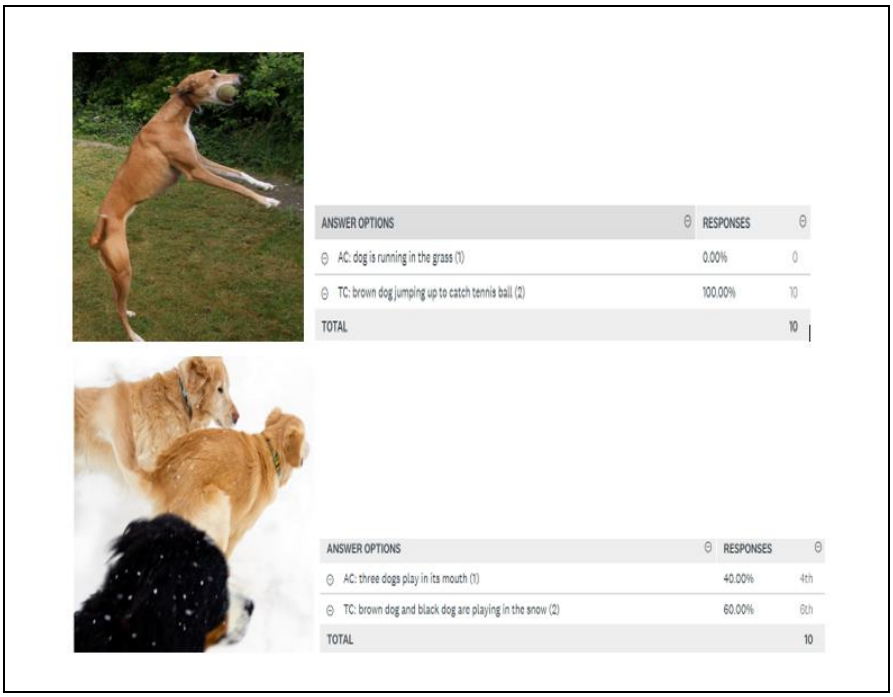


Figure 4.3 1 Human Survey Results (sample)

Most of the people chosen the caption generated by the Transformer model than the attention model, which once again proves our experiment that, transformer model performs better in image captioning than the Attention model.

Model	Human Response(Percentage)
Attention Model	25%
Transformer Model	75%

Table 4.3 1 Human Evaluation on image captions generated by Transformer and Attention Model

Human evaluation results from the survey output clearly shows that captions generated by the transformer model is better compared to the attention model.

4.4 Discussion

In this section will be discussion the strengths and weakness of the evaluation metrics used for this experiment.

4.4.1 Strengths

- **Ease of use:** BLEU metric is most widely used metric for evaluation, though its original purpose is for machine translation not for image captioning, it is still considered for image captioning, as it is efficient in finding the correlation between translated statement that is to be evaluated and reference statement.
- **Granularity:** Granularity is considered in an n-gram rather than a word, which is considered in longer matching information (Wang et al., 2021)
- **Quick Calculation:** As BLEU metric is based on calculating the precision of n-gram in a sentence, it is easy and fast to calculate

4.4.2 Weakness

- BLEU metric does not consider the meaning of the word, sometimes people use different word based on location and region for example lift can also be considered as elevator, but BLEU score will be reduced considering that to be incorrect
- It penalises on irrelevant words like “to”, “an” just as heavily as word that actually contributes significant meaning to the sentence.
- BLEU score always looks for exact word, when there is variant words like run, running it reduces the score
- Word order is not considered in BLEU score metrics for example “school is closed because of rain” is completely different from “rain is closed because of school”, but BLEU score will be 1 as it will get same Unigram, this is the major drawback in BLEU metrics.

5 CONCLUSION

5.1 Research Overview

The main objective of this research is to investigate image captioning by simple encoder-decoder model, Attention model and Transformer model, once generating the model, need to evaluate the best model among Attention and transformer model using statistical t-test on the BLEU metric scores obtained from each model and also performed Human evaluation for transparent evaluation.

5.2 Problem Definition

Many approaches have been proposed for image captioning which have been discussed in Chapter 2. The previous researchers and approaches have proposed different image captioning methods using template based, retrieval based methods or simple encoder-decoder architecture using CNN-LSTM model which have few issues and does not consider the spatial features and captions are generated as a whole. And with the help of attention mechanism, weights are added to the parameters having context vector with fixed-length would fail to retain longer sequences, henceforth transformer architecture has been used for this experiment to overcome such issues where input words are passed to encoder stack in parallel and uses multi-head attention to keep track of position of the input words.

5.3 Design/Experimentation, Evaluation and Results

The dataset Flickr8k has been gathered to perform the image captioning using simple encoder-decoder model, attention model and Transformer model. The dataset containing 8000 images and its captions has been used for this experiment. Data cleaning and pre-processing has been performed, hyperparameters of the model were defined and trained. Three different models has been created one is with simple CNN-LSTM model, where image captioning is done using greedy and beam search method, next for attention model Bahdanau et al. (2016) proposed attention mechanism has been adopted and the Transformers with multi-head attention has been implemented.

Once designing all these models, the performance of the attention and transformer model is analysed using BLEU metrics and statistical t-test is implemented with the BLEU scores obtained from each model for hypothesis testing. Furthermore, to have transparent analysis human evaluation has been performed by conducting a survey, in which images along with the captions generated by attention model and by transformer model has been presented.

Evaluation has been performed on the test set of the data and mean BLEU scores has been obtained from attention and transformer model. From the results obtained from Table 4.1-1, it shows that Transformer has outperformed Attention model in image captioning with mean BLEU score of 0.418. Furthermore, statistical t-test has been performed with the BLEU scores obtained from each model and p value is obtained from t-test is 0.007 which is less than 0.05 (significance level). Based on statistical test Null hypothesis is rejected. In previous sections results obtained from the model has been discussed with few of the images generated by attention model and transformer model. In few cases, though the BLEU score was less, the captions generated by the model was much accurate and appropriate, because BLEU score looks for the exact word match and penalises heavily on irrelevant words which does not make significant difference to the sentence. Therefore, reaching to a conclusion on this experiment just with BLEU score will not be enough so, to perform a fair analysis human evaluation was also performed, in which 75% have preferred the caption generated by the transformer than the attention model.

5.4 Contribution and Impact

Most aspects of image caption generation task has been compiled and model framework proposed in recent years has been discussed. For this research novel image captioning methods has been adopted which is based on deep learning models, like simple encoder-decoder model, attention model and transformer model. As there are many papers related to image captioning based on CNN-LSTM and its BLEU score, for this experiment have considered only attention model and transformer model for evaluation. Even after implementing the novel image captioning method like attention and transformer method, there is still an other areas of improvement required for the efficient image caption captioning and its evaluation metrics. In section 3.6 and 4.4 have discussed about the strengths and weakness of the model and the evaluation

metrics, therefore it is always more reliable and advisable to implement different methods and evaluate the results with different metrics.

From the table 4.1-1 and 4.1-2 it shows that Transformer model has performed comparatively well and be used efficiently in various fields where image captioning plays a major role like it could help visually impaired people or old aged people who have difficulty in recognizing the pictures, it could also be used in automotive industries for self-driving cars to indicate the traffic signals and for identifying the empty car parks during parking. Image captioning can also be used in medical fields for writing reports based on X-ray images.

5.5 Future Work and Recommendation

Although image captioning methods can be implemented in variety of fields the experimental results shows that there are still few areas of improvement for better performance.

- To improve efficiency of model, in future will make use of larger dataset like MSCOCO or Flickr30K dataset
- Implementation of different attention mechanism like adaptive attention and semantic attention would further improve the model
- Alternate architecture can be implemented for feature extraction like Xception or VGG19 for better performance.
- Use of different evaluation metrics like METEOR score, CIDEr or ROGUE could also give more elaborated results than just with one evaluation metric.

BIBLIOGRAPHY

Anderson, P., Fernando, B., Johnson, M., & Gould, S. (2016). SPICE: Semantic Propositional Image Caption Evaluation. *ArXiv:1607.08822 [Cs]*. http://arxiv.org/abs/1607.08822

Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., & Zhang, L. (2018). Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6077–6086. https://doi.org/10.1109/CVPR.2018.00636

Bahdanau, D., Cho, K., & Bengio, Y. (2016). Neural Machine Translation by Jointly Learning to Align and Translate. *ArXiv:1409.0473 [Cs, Stat]*. http://arxiv.org/abs/1409.0473

Bengio, S., Vinyals, O., Jaitly, N., & Shazeer, N. (2015). Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks. *In NIPS,2015*, 9.

Biswas, R., Barz, M., & Sonntag, D. (2020). Towards Explanatory Interactive Image Captioning Using Top-Down and Bottom-Up Features, Beam Search and Re-ranking. *KI - Künstliche Intelligenz*, *34*(4), 571–584. https://doi.org/10.1007/s13218-020-00679-2

Chen, L., Zhang, H., Xiao, J., Nie, L., Shao, J., Liu, W., & Chua, T.-S. (2017). SCA-CNN: Spatial and Channel-Wise Attention in Convolutional Networks for Image Captioning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6298–6306. https://doi.org/10.1109/CVPR.2017.667

Chu, Y., Yue, X., Yu, L., Sergei, M., & Wang, Z. (2020). Automatic Image Captioning Based on ResNet50 and LSTM with Soft Attention. Wireless Communications and Mobile Computing, 2020, 1–7. https://doi.org/10.1155/2020/8909458

Cornia, M., Stefanini, M., Baraldi, L., & Cucchiara, R. (2020). Meshed-Memory Transformer for Image Captioning. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10575–10584. https://doi.org/10.1109/CVPR42600.2020.01059

Doshi, K. (2021, May 21). *Image Captions with Deep Learning: State-of-the-Art Architectures*. Medium. https://towardsdatascience.com/image-captions-with-deep-learning-state-of-the-art-architectures-3290573712db

Dietterich, T. G. (1998). Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation*, *10*(7), 1895–1923. https://doi.org/10.1162/089976698300017197

Dubey, S., Olimov, F., Rafique, M. A., Kim, J., & Jeon, M. (2021). Label-Attention Transformer with Geometrically Coherent Objects for Image Captioning. *ArXiv:2109.07799 [Cs]*. http://arxiv.org/abs/2109.07799

He, S., Liao, W., Tavakoli, H. R., Yang, M., Rosenhahn, B., & Pugeault, N. (2020). Image Captioning through Image Transformer. *ArXiv:2004.14231 [Cs]*. http://arxiv.org/abs/2004.14231

Hodosh, M., Young, P., & Hockenmaier, J. (2013). Framing Image Description as a Ranking Task Data, Models and Evaluation Metrics Extended Abstract. *Journal of Artificial Intelligence Research 47*, 853–899

Hossain, MD. Z., Sohel, F., Shiratuddin, M. F., & Laga, H. (2019). A Comprehensive Survey of Deep Learning for Image Captioning. *ACM Computing Surveys*, *51*(6), 1–36. https://doi.org/10.1145/3295748

Hossain, M. Z., Sohel, F., Shiratuddin, M. F., Laga, H., & Bennamoun, M. (2019). Bi-SAN-CAP: Bi-Directional Self-Attention for Image Captioning. *2019 Digital Image*

Computing: Techniques and Applications (DICTA), 1–7. https://doi.org/10.1109/DICTA47822.2019.8946003

Hutchison, D., Kanade, T., Kittler, J., Kleinberg, J. M., Mattern, F., Mitchell, J. C., Naor, M., Nierstrasz, O., Pandu Rangan, C., Steffen, B., Sudan, M., Terzopoulos, D., Tygar, D., Vardi, M. Y., Weikum, G., Farhadi, A., Hejrati, M., Sadeghi, M. A., Young, P., … Forsyth, D. (2010). Every Picture Tells a Story: Generating Sentences from Images. In K. Daniilidis, P. Maragos, & N. Paragios (Eds.), *Computer Vision – ECCV 2010* (Vol. 6314, pp. 15–29). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-15561-1_2

Karpathy, A., Joulin, A., & Fei-Fei, L. (2014). Deep Fragment Embeddings for Bidirectional Image Sentence Mapping. *ArXiv:1406.5679 [Cs]*. http://arxiv.org/abs/1406.5679

Kiros, R., Salakhutdinov, R., & Zemel, R. (2014). Multimodal Neural Language Models. *IN ICML,2014*, 9.

Kulkarni, G., Premraj, V., Ordonez, V., Dhar, S., Li, S., Choi, Y., Berg, A. C., & Berg, T. L. (2013). BabyTalk: Understanding and Generating Simple Image Descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(12), 2891–2903. https://doi.org/10.1109/TPAMI.2012.162

Lei, Z., Zhou, C., Chen, S., Huang, Y., & Liu, X. (2020). A Sparse Transformer-Based Approach for Image Captioning. *IEEE Access*, *8*, 213437–213446. https://doi.org/10.1109/ACCESS.2020.3024639

Li, G., Zhu, L., Liu, P., & Yang, Y. (2019). Entangled Transformer for Image Captioning. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 8927–8936. https://doi.org/10.1109/ICCV.2019.00902

Liu, W., Chen, S., Guo, L., Zhu, X., & Liu, J. (2021). CPTR: Full Transformer Network for Image Captioning. *ArXiv:2101.10804 [Cs]*. http://arxiv.org/abs/2101.10804

Luong, M.-T., Pham, H., & Manning, C. D. (2015). Effective Approaches to Attention-based Neural Machine Translation. *ArXiv:1508.04025 [Cs]*. http://arxiv.org/abs/1508.04025

Mun, J., Cho, M., & Han, B. (2016). Text-guided Attention Model for Image Captioning. *ArXiv:1612.03557 [Cs]*. http://arxiv.org/abs/1612.03557

Ordonez, V., Kulkarni, G., & Berg, T. L. (2011). Im2Text: Describing Images Using 1 Million Captioned Photographs. *In Advances in Neural Information Processing Systems*, 1143–1151.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2001). BLEU: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL '02*, 311. https://doi.org/10.3115/1073083.1073135

Roy, A. (2020, December 9). *A Guide to Image Captioning*. Medium. https://towardsdatascience.com/a-guide-to-image-captioning-e9fd5517f350

Rush, A. M., Chopra, S., & Weston, J. (2015). A Neural Attention Model for Abstractive Sentence Summarization. *ArXiv:1509.00685 [Cs]*. http://arxiv.org/abs/1509.00685

Shen, X., Liu, B., Zhou, Y., Zhao, J., & Liu, M. (2020). Remote sensing image captioning via Variational Autoencoder and Reinforcement Learning. *Knowledge-Based Systems*, *203*, 105920. https://doi.org/10.1016/j.knosys.2020.105920.

Vedantam, R., Zitnick, C. L., & Parikh, D. (2015). CIDEr: Consensus-based image description evaluation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4566–4575. https://doi.org/10.1109/CVPR.2015.7299087

Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2017). Show and Tell: Lessons Learned from the 2015 MSCOCO Image Captioning Challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(4), 652–663. https://doi.org/10.1109/TPAMI.2016.2587640

Wang, C., Shen, Y., & Ji, L. (2021). Geometry Attention Transformer with Position-aware LSTMs for Image Captioning. *ArXiv:2110.00335 [Cs]*. http://arxiv.org/abs/2110.00335

Wang, H., Zhang, Y., & Yu, X. (2020). An Overview of Image Caption Generation Methods. *Computational Intelligence and Neuroscience*, *2020*, 1–13. https://doi.org/10.1155/2020/3062706

Wei, Y., Wang, L., Cao, H., Shao, M., & Wu, C. (2020). Multi-Attention Generative Adversarial Network for image captioning. *Neurocomputing*, *387*, 91–99. https://doi.org/10.1016/j.neucom.2019.12.073

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R., & Bengio, Y. (2016). Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. *ArXiv:1502.03044 [Cs]*. http://arxiv.org/abs/1502.03044

Xu, N., Zhang, H., Liu, A.-A., Nie, W., Su, Y., Nie, J., & Zhang, Y. (2020). Multi-Level Policy and Reward-Based Deep Reinforcement Learning Framework for Image Captioning. *IEEE Transactions on Multimedia*, *22*(5), 1372–1383. https://doi.org/10.1109/TMM.2019.2941820

Xu, Y., Wei, H., Lin, M., Deng, Y., Sheng, K., Zhang, M., Tang, F., Dong, W., Huang, F., & Xu, C. (2021). Transformers in computational visual media: A survey. *Computational Visual Media*, *8*(1), 33–62. https://doi.org/10.1007/s41095-021-0247-3

Y Yan, S., Wu, F., Smith, J. S., Lu, W., & Zhang, B. (2018). Image Captioning using Adversarial Networks and Reinforcement Learning. *2018 24th International*

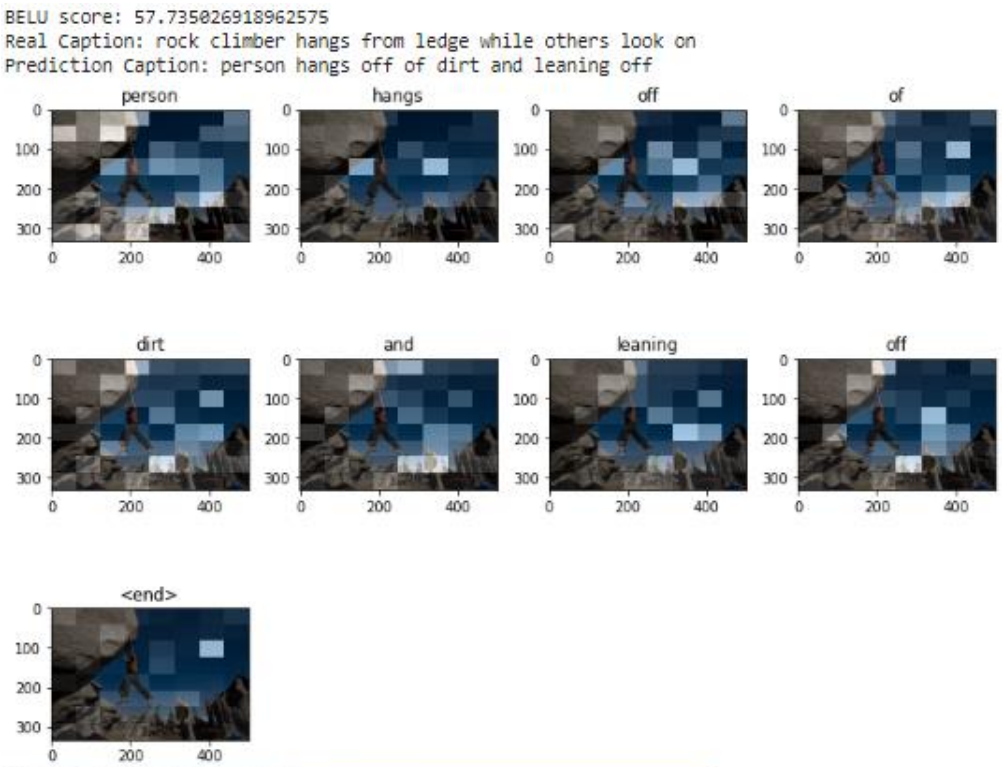
Conference on Pattern Recognition (ICPR), 248–253. https://doi.org/10.1109/ICPR.2018.8545049

Zheng, J., Krishnamurthy, S., Chen, R., Chen, M.-H., Ge, Z., & Li, X. (2019). Image Captioning with Integrated Bottom-Up and Multi-level Residual Top-Down Attention for Game Scene Understanding. *ArXiv:1906.06632 [Cs]*. http://arxiv.org/abs/1906.06632

APPENDIX A

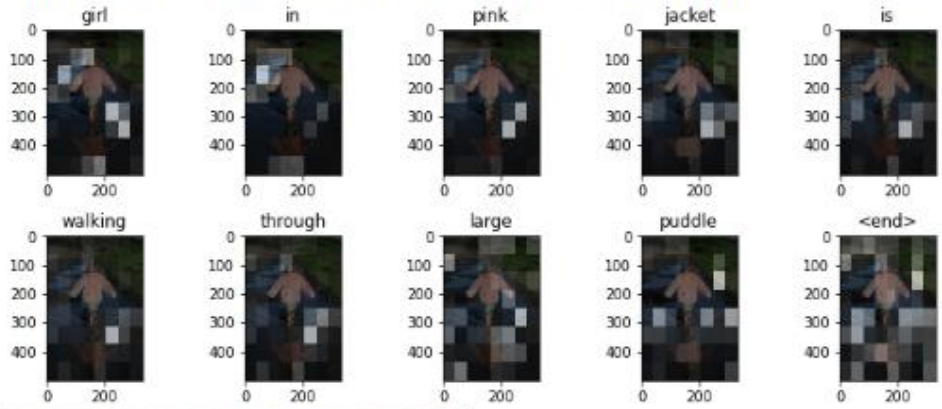
Few Examples of Image captions generated by Attention Model and Transformer model

PARTII: random output



Part2:O2

```
BELU score: 29.41733261715515
Real Caption: the girl in pink coat and pink <unk> is wading through puddle
Prediction Caption: girl in pink jacket is walking through large puddle
```

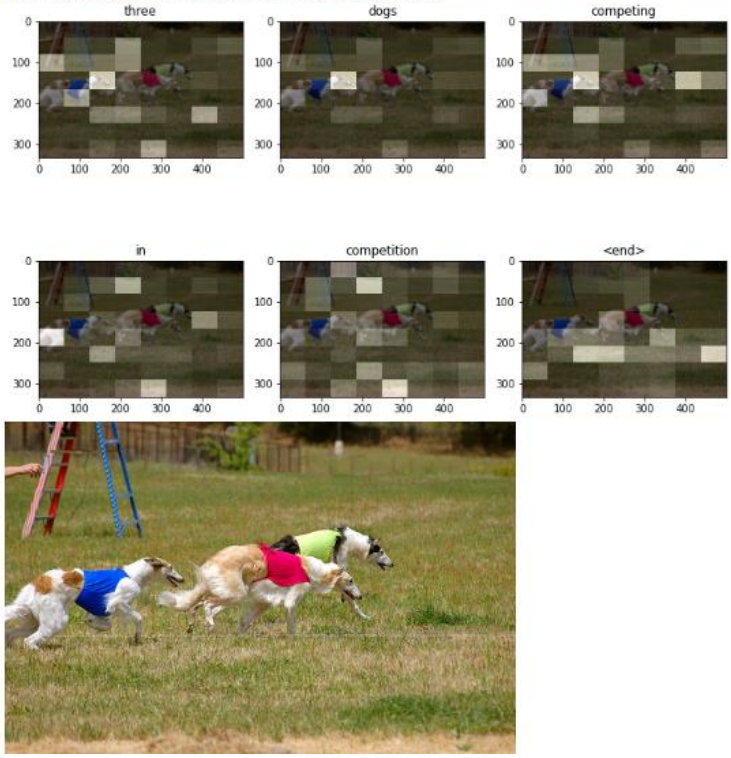


Part2:03

BELU score: 36.400302308068725
Real Caption: two girls are sitting in front of statue
Prediction Caption: woman are sitting on curb

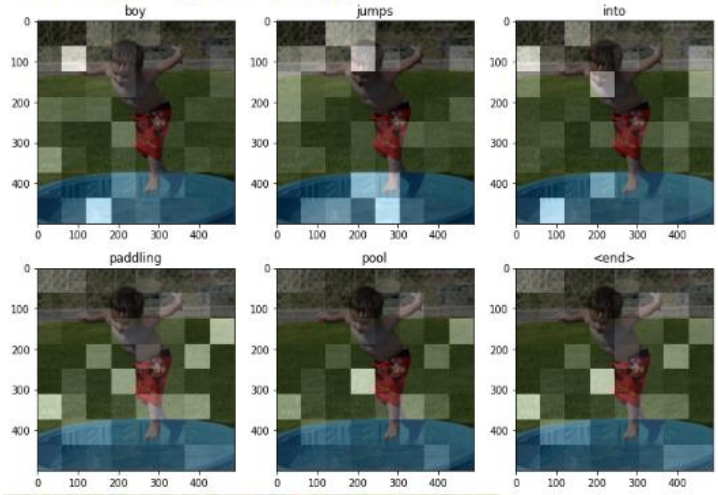
Part2:04

BELU score: 14.823156396438122
Real Caption: three dogs in colored sweaters run across grass with ladder and hand in background
Prediction Caption: three dogs competing in competition



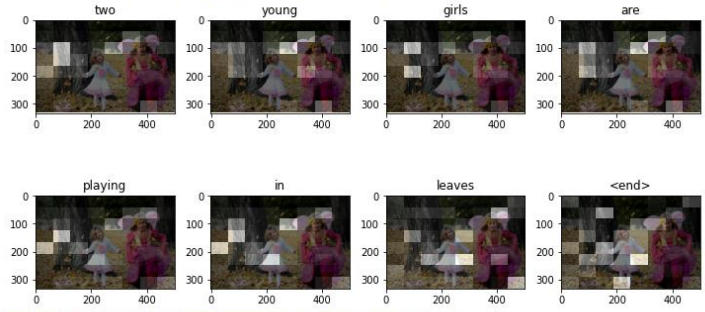
Part2:05

BELU score: 46.08636396914616
Real Caption: little boy in red trunks plays in kiddie pool
Prediction Caption: boy jumps into paddling pool



Part2:06

BELU score: 51.0029457493824
Real Caption: little girls who are dressed up play in the falling autumn leaves
Prediction Caption: two young girls are playing in leaves



Part3:01:

```
BLEU-3 score: 51.728185797171866
Real Caption: blond dog playing with colorful dog toy
Predicted Caption: dog in two looking at yellow chew chew bags
<matplotlib.image.AxesImage at 0x7f5854985a90>
```

Part3:o2:

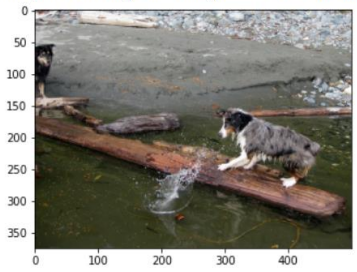
```
BLEU score: 62.993822983523565
Real Caption: dog running in the surf
Predicted Caption: the large brown and black dog is running through the water at the beach
<matplotlib.image.AxesImage at 0x7f5854b74f50>
```

Part3:03:

```
BLEU score: 55.778982530324605
Real Caption: two men doing some jumps outside
Predicted Caption: man doing skateboard trick on half pipe
<matplotlib.image.AxesImage at 0x7f5854c8e250>
```

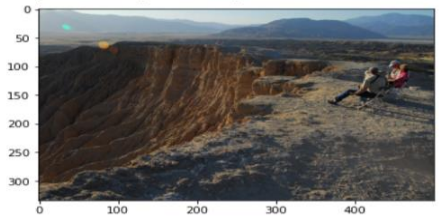
Part3:04:

```
BLEU score: 12.557058860058563
Real Caption: dog on log looks at splash in the water as another dog looks on
Predicted Caption: two dogs playing in water
<matplotlib.image.AxesImage at 0x7f5854cd8350>
```



Part3:05

```
BLEU score: 54.63634277011612
Real Caption: people in folding chairs take in the view from cliff
Predicted Caption: group of people sit on the edge of large rock while sitting on the ground
<matplotlib.image.AxesImage at 0x7f5854d11cd0>
```



Part3:06:

```
BLEU score: 32.503173264731565
Real Caption: brown dog leaping over an obstacle bar in room
Predicted Caption: brown dog jumps over an obstacle
<matplotlib.image.AxesImage at 0x7f5854e7af10>
```