# Image Caption Generation

Project report submitted for

**VI$^{th}$ Semester Minor Project**

Under the guidance of

**Dr. Vivek Tiwari**

By

**Amitesh Sahu (191020409)**

**Saswat Satapathy(191020449)**

**Setul Parmar (191020451)**


**Dr. Shyama Prasad Mukherjee**

**International Institute of Information Technology, Naya Raipur**

**(A Joint Initiative of Govt. of Chhattisgarh and NTPC)**

# CERTIFICATE

This is to certify that the project titled “IMAGE CAPTION GENERATION” by “Amitesh Sahu, Setul Parmar, Saswat Satapathy” has been carried out under my/our supervision and that this work has not been submitted elsewhere for a degree/diploma.

(Signature of Guide)

_____

**DR.VIVEK TIWARI**

**ASSOCIATE PROFESSOR**

**Department of _____CSE_____**

**Dr. SPM IIIT-NR**

**MAY, 2022**

# Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature of Author)

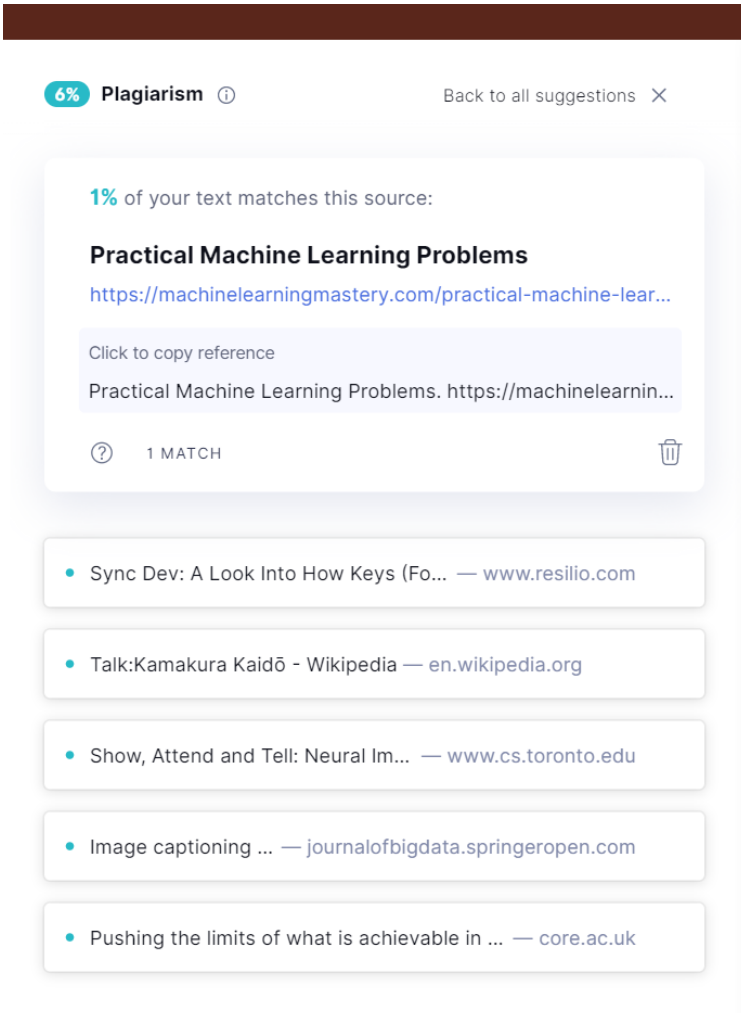
_____

**Amitesh Sahu (191020409)**

**Saswat Satapathy(191020449)**

**Date : _____** **Setul Parmar (191020451)**

# Plagiarism


6% Plagiarism
Back to all suggestions
1% of your text matches this source:
Practical Machine Learning Problems
https://machinelearningmastery.com/practical-machine-lear...
Click to copy reference
Practical Machine Learning Problems. https://machinelearnin...
1 MATCH
Sync Dev: A Look Into How Keys (Fo... — www.resilio.com
Talk:Kamakura Kaidō - Wikipedia — en.wikipedia.org
Show, Attend and Tell: Neural Im... — www.cs.toronto.edu
Image captioning ... — journalofbigdata.springeropen.com
Pushing the limits of what is achievable in ... — core.ac.uk

# Approval Sheet

This project report entitled “IMAGE CAPTION GENERATION REPORT” by “AMITESH, SETUL AND SASWAT” is approved for VI$^{h}$ Semester Minor Project.

(Signature of Examiner - I)

_____

Dr. Abhishek Sharma

(Signature of Examiner - I)

_____

Prof. Rajesh Ingle-II

(Signature of Examiner - II)

_____

Dr. Shailesh Khapre-III

(Signature of Chair)

_____

Dr.Vivek Tiwari

Date: _____ Place: _____

# Table of Contents

# List of All Tables

# List of All Figures

# ABSTRACT

With the advancement of deep learning, the combination of computer vision and natural language processing has gained considerable interest in recent years. Image captioning is an example of this field, in which the computer learns to grasp the visual information of an image by using one or more phrases. The process of generating meaningful descriptions for high level picture semantics includes not only the recognition of the item and the scene, but also the capacity to analyse the state, properties, and relationships between these things. Despite the fact that picture captioning is a demanding and tough process, several academics have made great progress. But none of the proposed works are enough to match and mimic human image understanding.

The image captioning task normalises object recognition to single-word descriptions. Most recent picture captioning research has concentrated on deep learning approaches, particularly Encoder-Decoder models incorporating Convolutional Neural Network (CNN) feature extraction. However, few efforts have attempted to improve the quality of produced captions by employing object recognition characteristics.

In our work we are using attention based Encoder-Decoder deep learning architecture. In the encoder part we are using CNN model pre-trained on ImageNet(XceptionNet) for extracting the spatial features of the image. We are also using the YOLOv4 model for extracting object features of the image. We then combine the spatial and object features and pass it to the decoder. An attention module (Bahdanau attention), a GRU, and two fully linked layers are used to generate language. We are using the famous Flickr8K dataset. We have used the BLEU Score as our evaluation metric. Our model is easy to train and analyse, and it creates captions by paying attention.

# 1. INTRODUCTION

In recent years, the idea of autonomously developing descriptive words for images has sparked attention in natural language processing and computer vision research. Image captioning is an important endeavour that requires semantic understanding of images as well as the ability to construct correct and exact description phrases.

Images are one of the most widely available data kinds on the Internet in the big data era, and the necessity for annotating and categorising them has grown. As an example of a big data problem, image captioning systems focus on the volume component of large data. The MS COCO dataset, for example, has approximately 123,000 images (25 GB). This necessitates the optimal use of resources as well as the careful planning of tests.

There are many important applications of image captioning. The picture captioning paradigm streamlines and automates the closed captioning process for digital content creation, editing, delivery, and archive. In order to generate quality captions for photographs and videos, well-trained models replace manual efforts. It is beneficial to visually challenged people as image descriptions can be read out to them allowing them to get a better feel of their surroundings. Tens of thousands of visual data points are distributed across borders by the media and public relations industries in the form of newsletters, emails, and so on. The picture captioning model speeds up subtitle generation and frees up executives' time for other vital responsibilities. Artificial intelligence is progressing from discussion rooms to underlying systems for finding and describing gigabytes of media content in social media. It allows administrators to monitor interactions and analysts to develop corporate plans.

Humans can easily understand image content and express it in the form of natural language sentences based on specific needs for image captioning; however, computers require the integrated use of image processing, computer vision, natural language processing, and other major areas of research results. The objective of picture captioning is to create a model that can fully utilise image data to generate more human-like rich image descriptions. The meaningful description generation process of high level image semantics necessitates not only an understanding of the image's objects or scene recognition, but also the ability to analyse their states, comprehend their relationships, and generate a semantically and syntactically correct sentence. It is currently unknown how the brain comprehends an image and organises visual data into a caption. Image captioning necessitates a thorough comprehension of the world and which elements stand out.

Early efforts relied on template methods, attempting to fill specified text templates with information retrieved from photos. Current systems take advantage of available processing power and employ deep learning techniques.

Implementing an Encoder-Decoder architecture is one of the most successful ways for image captioning. It encodes images to a high-level representation, which is subsequently decoded using a language generation model such as Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), or one of their variants. CNN are generally used as encoders.

The attention mechanism has proven to be useful in sequence-to-sequence applications, particularly picture captioning and machine translation. It improves accuracy by compelling the model to focus on the most critical bits of the input when creating output sequences.

Many recent deep learning models use pre-trained Convolutional Neural Networks (CNNs) to extract feature matrices from the final convolutional layers in order to analyse a picture. This allows you to grasp numerous characteristics of the items and their interactions in the image and represent it at a higher level. Some recent research attempted to employ object features for image captioning. The YOLOv3, YOLOv4, and YOLO9000 are among the models utilised, and are noted for their speed, precision, and effectiveness in real-time applications.

The aim of our work is to mimic human image understanding. To achieve this we are using encoder- decoder architecture. We are combining the spatial features obtained after the images through CNN( XceptionNet) and Object features obtained using the YOLO model. We are also using the Importance Factor on the object features so that the features which are more important i.e, the features about which more accurate information is available are given more importance. Then in the decoder part we use Bahdanau attention along with GRU and fully connected layers for caption generation. The accuracy of the generated caption is likely to be more since we are identifying the objects from the object features and taking into consideration other factors like background and importance of features.

# 2. LITERATURE REVIEW

According to Yin and Ordonez, a sequence-to-sequence model is proposed in which an LSTM network encodes a succession of objects and their positions as an input sequence and an LSTM language model decodes this representation to generate captions. Their methodology employs the YOLO [8] object detection paradigm to extract item layouts (object categories and locations) from photos and improve caption accuracy. They also offer a variant that extracts visual features using the VGG image classification model pre-trained on ImageNet. At each time step, the encoder receives a pair of object categories (encoded as a one-hot vector) and a location configuration vector including the left-most position, top-most position, width and height of the bounding box corresponding to the item, all normalised. The model is trained using back-propagation, however the mistake is not transferred to the object detection model. They demonstrated that when integrated with CNN and YOLO modules, their model improved in accuracy. They did not use all of the available data from the YOLO object attributes, such as object dimensions and confidence.

Vo-Ho et al. created an image captioning system that uses YOLO9000 and Faster R-CNN to extract object characteristics. Each type of feature is processed by an attention module to generate local features that reflect the section of the model that is presently being focused on. The probabilities of the words in the vocabulary set are generated at each time step by combining the two local feature sets and feeding them into an LSTM model. To process the findings, a beam search approach is utilised to select the best candidate caption. They extracted features from photos using the ResNet [14] CNN. They used YOLO9000 to extract a list of tags from a particular image, then broke each tag into words and eliminated repetitive ones so the list contained only unique terms. Each word I, including the "null" token, is represented as a single-hot vector of the vocabulary set's size. Then, using the word embedding approach, they embed each word into a d-dimensional space. For language generation, they employed LSTM units. Only the top 20 tags with the highest probability are retained.

Lanzendörfer et al. suggested a iBOWIMG-based paradigm for Visual Question Answering (VQA) in their paper. The model leverages the attention mechanism to collect features from Inception V3 as well as object characteristics taken from the YOLO [8] object detection model. In order to provide more meaningful characteristics to the iBOWIMG model, the YOLO outputs are encoded as vectors of size 80 1 with each column representing the number of detected objects of the specified category. Three of these object vectors are generated at detection confidence levels of 25%, 50%, and 75%, and then concatenated with image and question characteristics.

Herdade et al. suggested a spatial attention-based encoder-decoder model in which information regarding the spatial connection between sensed items is explicitly included. An object detector was used to extract appearance and geometry information from all detected items in the image, followed by the Object Relation Transformer to create caption text. For object recognition and feature extraction, they employed Faster R-CNN with ResNet-101 as the foundation CNN. Using intermediate feature maps from the ResNet-101 as inputs, a

Region Proposal Network (RPN) constructs bounding boxes for item proposals. Non-maximum suppression is used to eliminate overlapping bounding boxes with an intersection-over-union (IoU) greater than 0.7. All bounding boxes with class prediction probabilities less than 0.2 are likewise deleted. They then mean-pool across the spatial dimension for each item bounding box to create a 2048-dimensional feature vector. The Transformer model is then fed these feature vectors.

Wang et al. investigated picture captioning from start to finish using highly interpretable representations derived from explicit object recognition. They conducted a thorough examination of the efficacy of several object detection-based cues for picture captioning. They observed that frequency counts, item size, and position are all relevant and help to improve the accuracy of the captions generated. They also observed that certain item categories had a stronger impact on picture description than others.

Sharif et al. proposed using the linguistic relationships between items in a picture to improve image captioning quality. They use "word embeddings" to capture word semantics and encapsulate object semantic relatedness. To capture the physical and semantic closeness of object pairings, the proposed model employs linguistically aware relationship embeddings. It also employs NASNet to extract the image's global semantics. As a consequence, actual semantic relations that are not visible in the visual content of a picture may be learnt, allowing the decoder to focus on the most relevant object relations and visual aspects, resulting in more semantically-meaningful captions.

Vari et al. looked at the feasibility of textual and visual modalities coexisting in the same embedding space. They suggested a strategy that takes use of the textual character of object identification labels as well as the expressiveness of visual object representations produced from them. They studied if anchoring the representations in the captioning system's word embedding space, rather than grounding words or sentences in their corresponding pictures, may increase the efficiency of the captioning system. The grounding techniques they provide ensure that the projected object characteristics and the term embedding space are mutually grounded.

Alkalouti and Masre presented an Encoder-Decoder architecture-based paradigm for automating video captioning. They begin by selecting the most significant frames from the movie and removing those that are unnecessary. The YOLO model was utilised to recognise objects in video frames, and an LSTM model was used to generate language.

On a dataset of chest X-ray images, Ke et al. assessed the feature extraction performance of 16 common CNNs. They discovered no correlation between ImageNet performance and medical picture dataset performance. For medical tasks, they discovered that the choice of CNN architecture effects performance more than the concrete model within the model family. They also discovered that ImageNet pre-training improves performance in all designs, with a smaller gain in larger architectures. They also discovered that ImageNet pre-training results in a statistically significant improvement in performance across designs, with a greater improvement for smaller architectures.

Xu et al. developed an innovative Anchor-Captioner approach. They began by picking noteworthy tokens that needed greater attention and utilising them as anchors. The appropriate sentences for each anchor were then aggregated to form the anchor-centered

graph (ACG). Finally, in order to increase the content variety of produced captions, they utilised multi-view caption creation based on several ACGs.

Verb-specific Semantic Roles (VSR) were proposed by Chen et al. as a novel Controllable Image Captioning (CIC) control signal. VSR is composed of a verb and certain semantic roles that represent a specific action as well as the responsibilities of the entities participating in it. They trained a Grounded Semantic Role Labeling (GSRL) model to find and ground all things linked with each VSR role. They then proposed a Semantic Structure Planner to learn human-like descriptive semantic structures (SSP). Finally, they generated the captions using a role-shift captioning model.

Cornia et al. proposed a novel framework for picture captioning that enables both grounding and controllability in order to create varied descriptions. Given a control signal in the form of a series or collection of picture regions, they created the relevant caption using a recurrent architecture that explicitly predicts textual chunks based on regions and complies to the control's limits. Experiments are run using Flickr30k Entities and COCO Entities, a more sophisticated version of COCO with semi-automated grounding annotations. Their findings demonstrated that the strategy yields cutting-edge results in terms of caption quality and variety for customizable picture captioning.

| **Dataset** | **Training Split** | **Validation Split** | **Testing Split** | **Total Images** |
|---|---|---|---|---|
| Flickr30k | 28K | 1K | 1K | 30K |
| MS COCO | 8.3K | 41K | 41K | 144K |

**FIGURE 2.1** A COMPARISON BETWEEN THE DATASETS

# 3. Datasets

We have used the Flickr 8k dataset in this project and have planned to use the MS COCO dataset and flickr30k as part of our future works.

## Flickr 8k:

It is a collection for sentence-based image description and search, containing 8,000 photos linked with five different captions that provide clear descriptions of the salient items and events. The photographs were chosen from six different Flickr groups and do not usually include well-known people or places, but were hand-picked to illustrate a diversity of scenes and circumstances.

It contains the following files:

- **Flickr8k_Dataset :** There are a total of 8092 JPEG images in all forms and sizes. 6000 are for training, 1000 for validation, and 1000 for testing.
- **Flickr8k.token.txt :** the Flickr8k Dataset's raw captions. The caption's ID is "image address # caption number" in the first column.
- **Flickr8k.lemma.txt :** a lemmatized version of the captions above.
- **Flickr 8k.trainImages.txt :** The photographs utilized in our tests as training.
- **Flickr 8k.devImages.txt :** The photographs utilised in our experiments for development and validation.
- **Flickr 8k.testImages.txt :** These are the test photographs that we used in our experiments.
- **ExpertAnnotations.txt :** contains expert opinions. The image and caption IDs are listed in the first two columns. The caption IDs are 0-4>. The expert opinions for that image-caption pair are presented in the following three columns. Scores range from 1 to 4, with 1 indicating that the caption does not describe the image at all, 2 indicating that the caption describes minor aspects of the image but does not describe the image, 3 indicating that the caption nearly describes the image with minor errors, and 4 indicating that the caption describes the image.
- **CrowdFlowerAnnotations.txt :** CrowdFlower judgments are present in it. The image and caption IDs are listed in the first two columns. The third column shows the percentage of Yeses, the fourth the total number of Yeses, and the fifth the total number of No. A Yes indicates that the caption describes the image (albeit with slight errors), whereas a No indicates that the caption does not describe the image. Each image-caption pair contains at least three judgments, but some may have more.

# 4. PROPOSED SOLUTION

The experimental method entails obtaining object features from the YOLO model and incorporating them into a simple deep learning model that employs the widely used Encoder-Decoder architecture with the attention mechanism, along with CNN convolutional features. The "Results and Discussion" section compares the results before and after the object features were added. Although prior research recorded object features as a vector, we made a significant improvement by simply concatenating them. We also examine the impact of ordering the YOLO object tags according to a metric that we suggest.

**Metrics for evaluation**

We employ a set of assessment measures that are extensively utilised in the field of picture captioning. In automated text evaluation, **BLEU** metrics are commonly used to quantify the correspondence between a machine translation output and a human translation; in the case of image captioning, the machine translation output corresponds to the automatically produced caption, and the human translation corresponds to the human description of the image.

**Bleu Score is given by:**

$$P(n) = \frac{C\varepsilon\{Candidates^{\Sigma}\}\, n-grams\, \varepsilon\, C^{\Sigma\, Count_{Clip}(n-gram)}}{C'\varepsilon\,\{Candidates^{\Sigma}\}n-gram'^{\Sigma}\varepsilon\, C'^{\,CountClip(n-gram')}}$$

**Model**

Our model uses an attention-based Encoder-Decoder architecture. It has two methods of feature extraction for image captioning: an image classification CNN (Xception), and an object detection model (YOLOv4).

The outputs of these models are concatenated to create a feature matrix that gives the language decoder more information to anticipate more accurate descriptions. We employ raw item layout information directly, unlike other works that incorporate object features before merging them with CNN features. An attention module (Bahdanau attention), a GRU, and two fully connected layers are used to generate language. Our approach creates captions utilising attention and is straightforward to train and analyse.

**Encoding of images**

A. CNN with pre-trained image classification To extract spatial characteristics, we use the Xception CNN, which has been pre-trained on ImageNet.

Xception (Extreme version of Inception) is based on Inception V3, but instead of modules, it contains 71 layers with a depth-wise separable convolution that may be modified. Because to superior model parameter utilisation, it outperforms Inception V3.

Following recent work in picture captioning, we extract characteristics from the last layer before the completely linked layer. Instead of relying solely on the picture class, this allows the whole model to acquire knowledge into the objects in the image and their relationships.

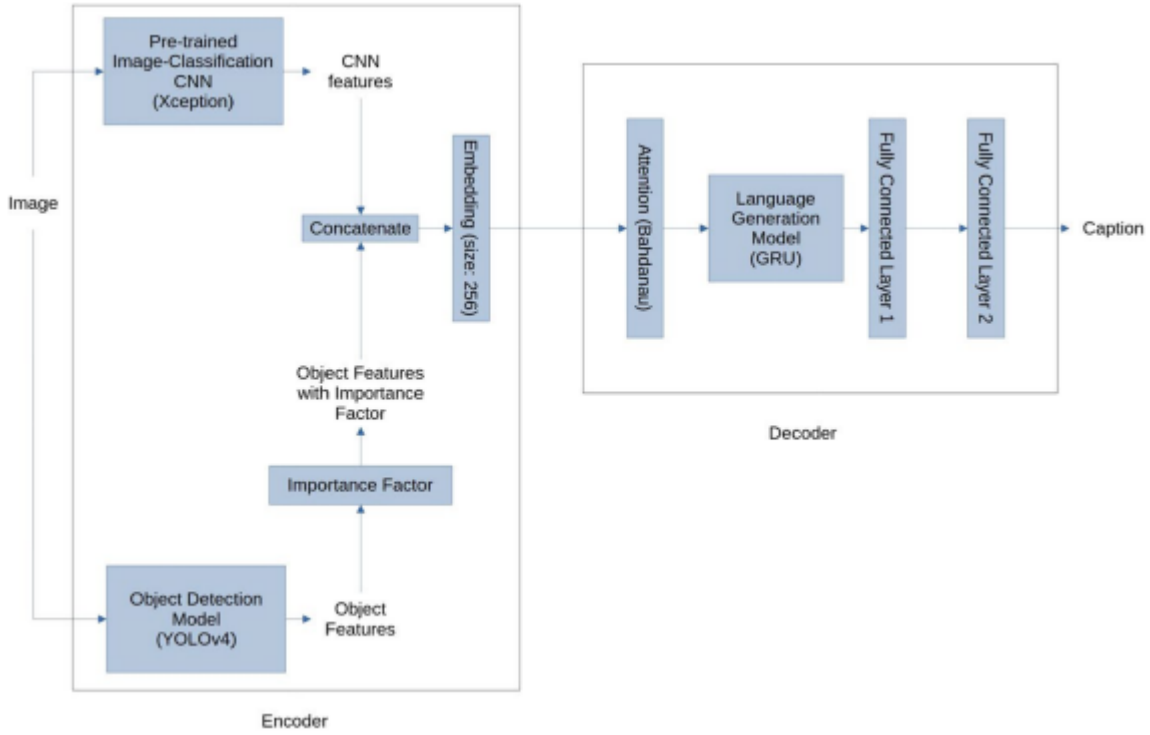



**FIGURE 3.1 BLOCK DIAGRAM OF THE PROPOSED SYSTEM**

Different feature extraction CNN models for picture captioning applications were compared in a prior study. The results showed that Xception was one of the most reliable feature extraction models, hence it was chosen as the feature extraction model in this study. This stage's output is of the shape, which was flattened for matrix handling.

The model summary without using object detection is given below:

```
Model: "cnn__encoder"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               multiple                  524544

=================================================================
Total params: 524,544
Trainable params: 524,544
Non-trainable params: 0
_________________________________________________________________
None


Model: "rnn__decoder"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       multiple                  1873664

 gru (GRU)                   multiple                  1575936

 dense_1 (Dense)             multiple                  262656

 dense_2 (Dense)             multiple                  3754647

 bahdanau_attention (Bahdana  multiple                 394753
 uAttention)

=================================================================
Total params: 7,861,656
Trainable params: 7,861,656
Non-trainable params: 0
_________________________________________________________________
None
```

**Model for detecting objects**

The YOLOv4 model is used in our method because of its speed and precision, making it suited for massive data and real-time applications. The extracted features are a list of object features, each of which contains the X and Y coordinates, width and height, confidence rate (from 0 to 1), class number, and a new optional "importance factor."

When describing an image, foreground things are typically larger and more essential, whereas background objects are typically smaller and less important. Furthermore, it makes more sense to employ more accurate data rather than less precise data.

**Importance Factor = Confidence Rate × Object Width × Object Height**

The features list is compressed into a 1D array with a length of less than 2048. It is then padded with zeros to make it compatible with the CNN module's output of 2048 bytes. This stage's result is an array (1 2048).

YOLO divides an image into a grid to calculate the confidence score. In each of these grid cells, bounding boxes and confidence scores for these boxes are anticipated. The confidence score indicates how confident the model is that the box contains an object and how accurate the model believes the predicted box is.

The detection of objects Intersection over Union (IoU) between the predicted box and the ground truth is used to evaluate the method. It examines how closely the anticipated box matches the ground truth by calculating the ground truth and anticipated bounding box overlap .If there is no object in the cell, the confidence score should be zero. The following is the formula for determining the confidence score:

C = Pr(object)* IoU

Embedding and concatenation :

We add this concatenation step to take advantage of the picture classification and object detection characteristics by attaching the output of the YOLOv4 subsystem as the last row in the output of stage 1. This stage's output is shaped (101*2048).

One fully connected layer of length 256 is used for the embedding. This stage maintains feature size consistency and translates the feature space to a smaller space suitable for the language decoder.

The attention method was developed to improve the encoder decoder architecture's performance in machine translation. Attention was also important for evaluating photos, as image captioning can be considered as a specific case of machine translation. By integrating all of the encoded input vectors into a weighted combination, with the most relevant vectors obtaining the largest weights, the attention mechanism was designed to allow the decoder to use the most relevant sections of the input sequence in a flexible manner.

When describing an image, attention follows the human instinct of focusing on different elements of it. Using object detection features is similarly based on the idea that knowledge about item classes and positions might help you understand the image better than just using convolutional features. When both feature types are used, the system will pay attention to different features of both object classes and positions in the same image.

**Language Decoder**

A GRU is used for decoding because of its speed and low memory utilisation. It generates a caption by producing one word at a time, based on a context vector, the prior hidden state, and previously generated words. The backpropagation algorithm is used to deterministically train the model.

Two completely connected layers follow the GRU. The first is 512 characters long, while the second is the amount of the vocabulary used to generate output text.

The decoder's training procedure is as follows:

1. The encoder is used to extract the features, which are then sent through.

2. The encoder output, hidden state (initialised to 0), and decoder input are sent to the decoder (which is the start token).

3. The decoder returns both the predictions and the decoder's concealed state.

4. The model is then updated with the decoder's hidden state, and the loss is determined using the predictions.

5. To decide the next decoder input, "teacher forcing" is used, which is a technique that involves feeding the decoder the target word as the next input.

**Pre-processing**

This section describes the data pre-processing algorithm that was used:

1. Sort the collection into image-caption pairs at random. This allows the training process to converge quickly and eliminates bias during the process. As a result, the model is unable to learn the training order.

2. Read the images and decode them.

3. Resize the photos to the CNN requirements: the image is resized to 299299 pixels, as required by the Xception CNN model.

4. The text is tokenized. Tokenization divides raw text into words separated by punctuation, special characters, and white spaces. The separators are thrown away.

5. Count the tokens, arrange them by frequency, and make the system's vocabulary out of the top 15,000 most common words. By removing terms that are unlikely to be relevant, this prevents overfitting.

6. Create index-to-word and word-to-index structures. After that, they're employed to convert token sequences into word identifier sequences.

7. Padding. Because sentences can vary in length, we need the inputs to be the same size, which is where the padding comes in. Identification sequences are padded with null tokens at the end to ensure that they are of the same length.

# 5. RESULTS

## 5.1 Without object detection

TensorFlow1 is used in our code, which is written in the Python programming language. Keras2 was used to import the CNN implementation and trained model and a library, as well as a pre-trained YOLOv4 model on MS COCO was imported from the yolov4 library. These Scores are calculated using the MS COCO evaluation tool.

Below each image given are the Real captions, the predicted captions and the corresponding Bleu Score.

```
Real Caption: ['black and white dog plays in the water', 'black and white dog walking in shallow water'
Prediction Caption: dog is swimming in water
BLEU score -> 0.8408964152537145
```

```
Real Caption: ['person in black wetsuit is surfing in the ocean with wave coming down',
Prediction Caption: surfer rides the waves chases while waves on his board and one hand
BLEU score -> 0.09429254736980938
```

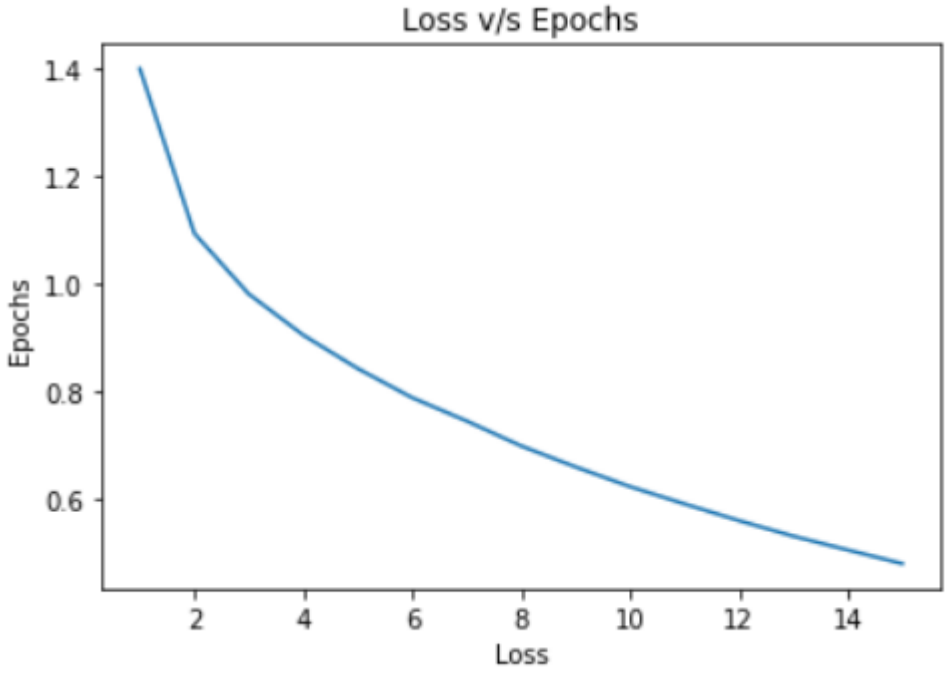
Two commonly used datasets for image caption creation are employed in the tests: MS COCO and Flickr30k. In these two datasets, which comprise 123,000 and 31,000 photos, respectively, each image has four reference captions. According to Karpathy's split, 5000 photos are reserved for validation and 5000 images are reserved for verification in MS COCO. 29,000 photos are utilised for preparation, 1000 for validation, and 1000 for testing in

the Flicker30k dataset. The loss function was Sparse Categorical Cross Entropy, and the model was trained for 1 epochs. Adam optimizer was used as the optimizer.


Real Caption: ['homeless man in white cap is standing behind large orange cones', '
Prediction Caption: the man is wearing white baseball hat
BLEU score -> 0.7825422900366437

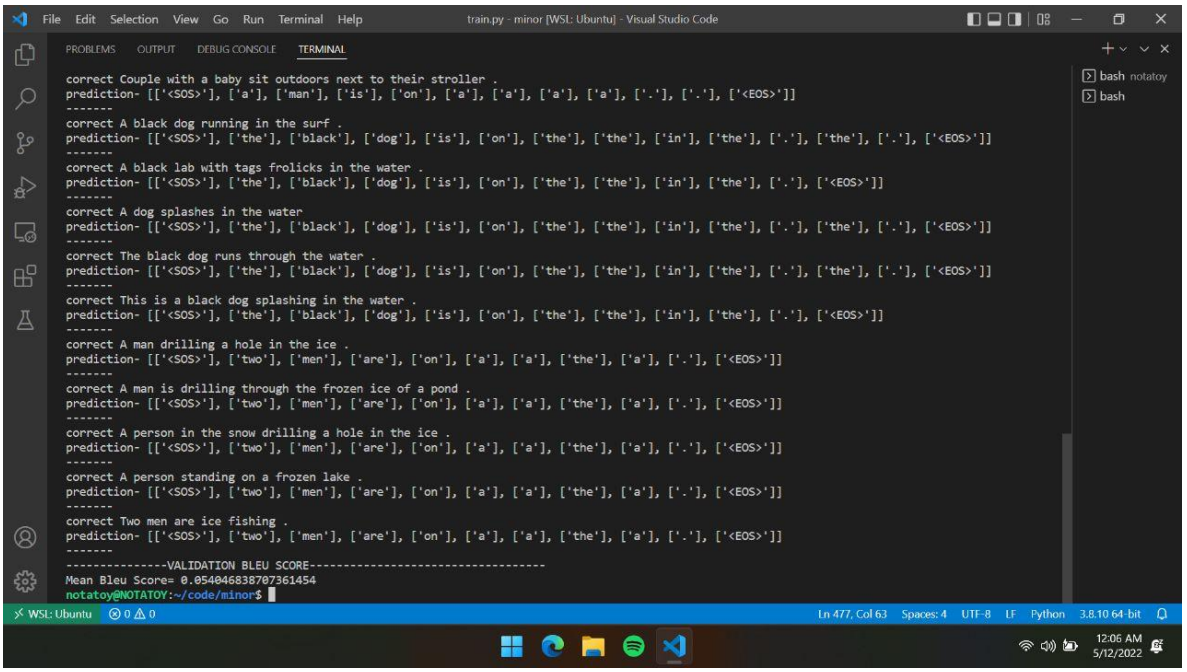
Real Caption: ['black and brown dog stands on roof', 'black dog at the top of red incline',
Prediction Caption: the dog has its tongue out
BLEU score -> 0.5169731539571706

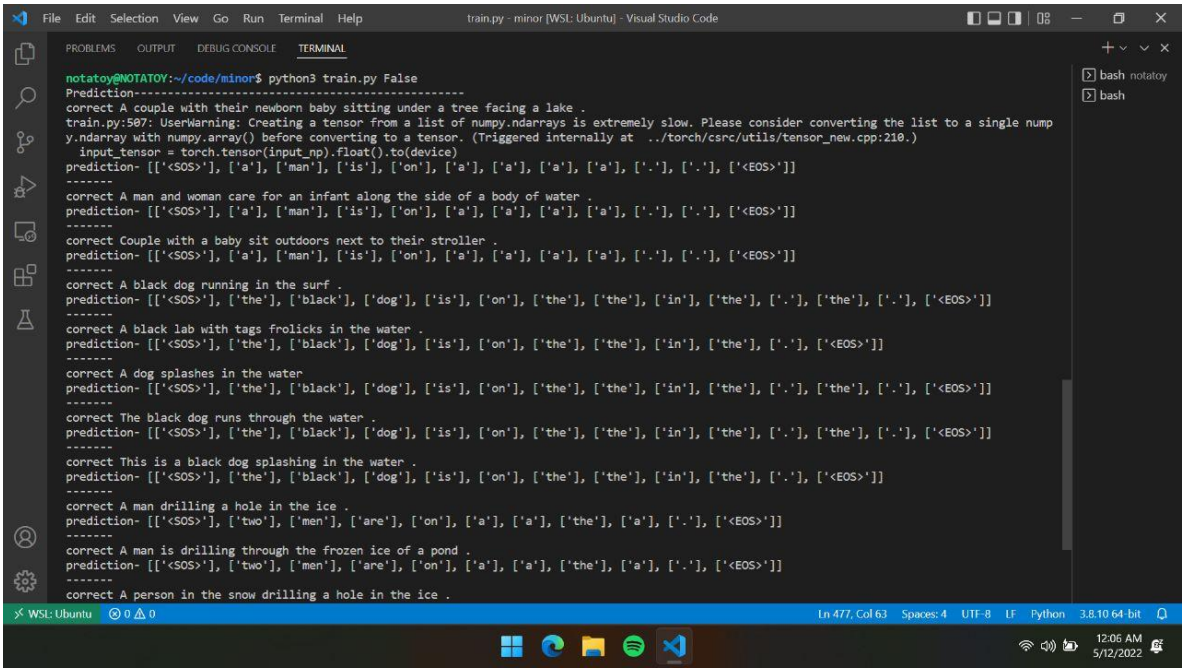
**Loss v/s Epochs graph**



# 5.2 With object detection

**Below images contain the correct captions as well as the predicted captions by our model.**



# 6. CONCLUSIONS

- The captions generated are sufficiently accurate when there is a single object in the image and the quality of captions decrease when the number of objects increase.

- Generating caption via object detection makes the result worse if the importance factor is not introduced and size of the dataset is less.

- BLEU score does not take into account the sequence of words and context of sentences. Thus, BLEU score is not the ideal metric for image caption generation evaluation.

- We demonstrated the efficacy of an attention-based Encoder-Decoder image captioning model that employs two methods of feature extraction, an image classification CNN (Xception) and an object identification module (YOLOv4).

- We introduced the significance factor, which favours high-confidence objects over low-confidence ones and prioritises foreground large objects over background little ones, and demonstrated how it affects increasing scores.

- We demonstrated how our strategy enhanced scores and compared it to earlier studies in terms of score increases, particularly the CIDEr measure, which climbed 15.04 percent, indicating improved grammatical saliency.

- Instead of only item layouts, future work could benefit from rich object semantic information from caption texts, which could improve image captioning precision.

- Furthermore, more advanced ways for encoding object features before feeding them into the decoder can be used, as well as more complex language models.

# 7. REFERENCES


1) Al-Malla, M.A., Jafar, A. & Ghneim, N. Image captioning model using attention and object features to mimic human image understanding. *J Big Data* **9**, 20 (2022)

2) Zhongliang Yang, Yu-Jin Zhang, Sadaqat ur Rehman, Yongfeng Huang.Image Captioning with Object Detection and Localization.

3) Detection and Recognition of Objects in Image Caption Generator System: A Deep Learning Approach June,2019

4) Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. Adv Neural Inf Process Syst. 2015;28:91–9.

5) Lanzendörfer L, Marcon S, der Maur LA, Pendulum T. YOLO-ing the visual question answering baseline. Austin: The University of Texas at Austin; 2018.

6) Herdade S, Kappeler A, Boakye K, Soares J. Image captioning: transforming objects into words. https://arxiv.org/abs/1906.05963. Accessed 14 Jun 2019.

7) Alkalouti HN, Masre MA. Encoder-decoder model for automatic video captioning using yolo algorithm. In: 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS). Piscataway: IEEE; 2021. p. 1–4.

8) Xu G, Niu S, Tan M, Luo Y, Du Q, Wu Q. Towards accurate text-based image captioning with content diversity exploration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE; 2021. p. 12637–46.

9) Chen L, Jiang Z, Xiao J, Liu W. Human-like controllable image captioning with verb-specific semantic roles. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE; 2021. p. 16846–56.

10) Papineni K, Roukos S, Ward T, Zhu WJ. Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting of the Association for Computational Linguistics. Philadelphia: ACL; 2002. p. 311–8.

11) Vo-Ho VK, Luong QA, Nguyen DT, Tran MK, Tran MT. A smart system for text-lifelog generation from wearable cameras in smart environment using concept-augmented image captioning with modified beam search strategy. Appl Sci. 2019;9(9):1886.

12) Redmon J, Farhadi A. Yolov3: an incremental improvement.

https://arxiv.org/abs/1804.02767. Accessed 8 Apr 2018.

13) ochkovskiy A, Wang CY, Liao HY. Yolov4: optimal speed and accuracy of object detection. https://arxiv.org/abs/2004.10934. Accessed 23 Apr 2020.

14) Redmon J, Farhadi A. YOLO9000: better, faster, stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway: IEEE; 2017. p. 7263–71.