Image Captioning using CNN and Attention Based Transformer

Deepa Mulimani, Prakashgoud Patil, Nagaraj Chaklabbi

KLE Technological University, Hubballi

Corresponding author: Deepa Mulimani, Email: 1deepamulimani19@gmail.com

Image captioning is a technique for generating sentences that describe a scenario captured in photos. It can identify objects in a picture and carries out a few processes with the goal of locating the image's most crucial parts. Algorithms now have the ability to generate text in the context of natural phrases that accurately describe an image. To extract image visual features, this work employs a pre-trained Convolution Neural Network (CNN) viz. EfficientNetB0, and then uses Transformer Encoder and Decoder to construct an appropriate caption. The model is trained using the Flickr8k dataset. The findings back up the model's capacity to understand and produce text from pictures. The evaluation metric is the BLEU (bilingual evaluation understudy) score. The model obtains the image description, converts into text, and then into a voice. For visually impaired people who are unable to grasp visuals, image description is the ideal approach.

Keywords: Image caption generator, CNN, Transformer, Attention, Decoder, Encoder, Efficient-Net.

2023. In Satyasai Jagannath Nanda & Rajendra Prasad Yadav (eds.), *Data Science and Intelligent Computing Techniques*, 157–166. Computing & Intelligent Systems, SCRS, India. https://doi.org/10.56155/978-81-955020-2-8-14

1 Introduction

A right description of an image is regularly stated as 'Visualizing a photo with inside the thoughts'. The advent of an image in thoughts can play an enormous position in sentence generation. This process of routinely generating captions describing the activity in the image is far more difficult than object recognition and image classification. An image's description must include not just include the image's most effective elements, but also the least successful ones and also the relation among the items with their attributes and activities shown in images.

In this era of artificial intelligence, it is vital for large corporations like Google to generate captions for photographs to enhance the search experience by enabling image searches. This type of image captioning algorithm is also utilized by social media sites such as Twitter, Facebook, Instagram, and Snapchat to forecast the user's potential interests and adjust the user's feed appropriately. Finding direct objects in an image is not a particularly tough process. However, it is difficult for machines to identify the image's prominent aspects, such as children playing on a playground. The resulting captions should be correct in terms of syntactic and semantic accuracy. The availability of large datasets for free, such as Flickr30k, Flickr 8k, ImageNet, and Microsoft COCO dataset has aided research on this subject. Furthermore, Neural Network inspired encoder-decoder structures, such as Recurrent Neural Networks and Convolutional Neural Networks, have greatly assisted research in this field.
Deep Learning methods are implemented using neural networks. This has a large use on social networking platforms, where users may submit photos and have them analyzed to produce captions. Additionally, the generated captions can assist to track which users enjoy certain types of content. This technology can also assist those with visual impairments with the scene-to-text conversion if the resulting text (caption) is turned to audio.

The results are compared with GABC and some more recent variants of ABC. The results are very promising and show that the proposed algorithm is a competitive algorithm in the field of swarm intelligence-based algorithms.

2 Related Work

Xinxin Zhu et al. presented the article "Captioning Transformer with Stacked Attention Modules". It describes a stacked attention module Captioning Trans-former (CT) concept. It presents a framework consisting of a CNN encoder, a stacked Transformer Decoder with multi-attention, and a feed-forward layer. Text, Image embedding, and Multi-Head Attention are discussed. On convention-al assessment measures, the Caption Transformer (CT) model was compared to several state-of-the-art methods such as ROUGE, BLEU-1, BLEU-2, BLEU-3, BLEU-4 and METEOR. Kelvin Xu et al. presented the paper "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention". This research offers an attention-based approach that learns to automatically characterize visu-al content and how it can be trained deterministically by classic back propagation methods are used, as well as stochastically maximizing a variation lower bound. It also shows how, via visualization, the model can learn to concentrate its eyes on crucial elements while constructing the proper words in the output sequence. It introduces two image caption generators that are based on attention (soft and hard Attention). To accomplish quantitative analysis, three benchmark datasets are utilized: Flickr8k, Flickr30k, and MS COCO, and BLEU and METEOR metrics are used to assess the performance of model out- puts. Andrej Karpathy et al. presented "Deep Visual-Semantic Alignments for Generating Image Descriptions" by proposing natural language descriptions of images and their areas. Their alignment methodology is based on Convolutional Neural Networks over image areas and bidirectional Recurrent Neural Networks over phrases through a multimodal embedding alignment to train to provide a description of image areas. Haoran Wang et al. presented a research article "An Overview of Image Caption Generation Methods". This article provides brief views on different methods for Image Feature Extraction models using neural networks and statistical language models with encoder and decoder models. Different types of attention mechanisms are explained and their Comparison is presented. The results of different attention models are compared using the BLEU score. Zelin Deng et al. presented the article "A Position-Aware Transformer for Image Captioning". This paper presents the Image-feature attention and position-aware attention methods that are used in the Position-Aware Transformer model. The use of Feature Pyramid Network (FPN) to extract multi-level features and the usage of scaled-dot-product to combine these qualities are detailed. The experimental findings of the Position-aware Transformer on the MS COCO testing set are compared to previous state-of-the-art models and reported. Jun Yu et al. presented the article "Multimodal Transformer with Multi-View Visual Representation for Image Caption-ing". This paper extends the existing transformer model to a Multimodal Trans-former (MT) for image caption generation. It shows the MT model's ability to perform complex reasoning and accurate output. It de- scribes the aligned and unaligned multi-view image encoder concepts. This paper has a large analysis of the attention model results. Shuang Liu et al. presented a research paper "Image Captioning Based on Deep Neural Networks". This research primarily discusses the three deep neural network- based CNN-CNN, CNN-RNN, and reinforcement-based frameworks that are used to caption images. Then the evaluation metrics, benefits, and significant problems for each of these three best techniques are discussed. Simao Herdade et al. presented the paper "Image Captioning: Trans-forming Ob- jects into Words". The Object Relation Transformer architecture is introduced in this paper, which extends the technique by explicitly adding information about the spatial connection between inputs identified items via geometric attention. The value of such geometric attention for picture captioning is demonstrated by quantitative and qualitative findings on the MS-COCO dataset, which show gains in all popular captioning criteria. Lakshminarasimhan Srinivasan et al. presented "Image Captioning - A Deep Learning Approach". This paper presents the LSTM model for Image Captioning. In this paper Image Feature Ex-traction, Sequence processor, and Decoder are briefly explained and the results are evaluated providing an overview of LSTM-based Image Captioning. Murk Chohan et al. presented Journal "Image Captioning using Deep Learning: A Systematic Literature Review". This paper explores about Image Caption in a way that how large a topic has grown, and the number of researches done on what kind of models, datasets, and their results. It gives a vast view of the Image Captioning re- search topic.

After surveying the articles, the numerous challenges like speed, accuracy, and training loss remain unresolved despite the accomplishments of many systems like Recurrent Neural Networks (RNN), LSTM, etc. This work focuses on improving accuracy, speed, and training loss. The main objective is to use the Flickr8K dataset to develop a deep learning model that will generate captions explaining the activity in the image and convert it into speech. This type of approach is useful for blind persons who can use voice to understand any visual

3 DATASET

3.1 Data Sources

The three most often used image caption training datasets in Computer Vision re- search are the MS COCO dataset, Flickr30k dataset, and Flickr8k dataset [10]. These datasets comprise (123,000), (31,000), and (8,000) captioned photos, with each image labeled with five different descriptions.

3.2 Flickr8k Dataset

Flickr8k is a publicly available benchmark dataset for image-to-sentence de-scription with 8000 image files, each with five captions. Each caption describes the entities and events depicted in the image in detail. The training dataset has 6000 photos and the testing dataset contains 2000 images. Multiple captions mapped to a single image make the model general and prevent overfitting. The given text captions are:

- A BMX rider in a red and black outfit is jumping on his motorcycle.
- A dirt biker flies through the air.
- A guy in red on a bike in mid-air.

- A man in a red outfit jumps his motocross bike down the hill

3.3 Data Pre-processing

There are two sorts of pre-processing needed: one for photos and one for text (captions) [9]. In the initial phase of image pre-processing, the images must be translated into an input format that is acceptable with the CNN's input. Addition-ally, the captions (text) must be associated with their appropriate image titles. Later, the captions are separated into individual words to make a dictionary. To con-struct vectors, the words within the dictionary are processed through tokenization. After tokenizing, the resultant vectors are padded to ensure that all tokens have the same size (word vectors). Image feature vectors and Caption vectors are paired and properly learned during training. Every image must be formatted into a fixed-size vector before being fed into the neural network. For this reason, we have chosen transfer learning, employing Google Brain's EfficientNet-B0 pre-trained network model (Convolutional Neural Network).

4 METHODOLOGY AND WORKING

4.1 Algorithm of the proposed model

Fig. 1 presents the algorithm of the proposed model and below is the procedure.

Step 1: Download the dataset

The Flicker8k dataset has 8000 images, and the document is formatted with an image and caption separated by a new line. i.e., it contains the image's name, followed by a space and its description in Text format. With the use of a dictionary, the images are mapped to their respective descriptions [12].

Step 2: Preparing the dataset

One of the fundamental steps in NLP is to reduce noise. Noise can take the form of discrete characters like hashtags, punctuation, and numbers. Generate the Vocabulary; Vocabulary is a collection of specific words found in the text corpus. Load the images to map them to their appropriate descriptions in the education set, which are stored under the description's variable.

Caption data are loaded and mapped to corresponding images using the path to the text document that contains the image names and 5 corresponding captions for Dictionary mappings. Each caption is given a start and an end token, and those that are too short or too long are removed. The mapped dataset is completely trained and it returns Training and validation dataset samples as two distinct dictionaries [12].

Step 3: Vectorizing the text data

Text data will be vectorized using the Text Vectorization layer, which will convert the original strings into integer sequences, in which each integer indicates the index of a word in the vocabulary. Custom splitting string standardization technique is utilized, i.e. one that strips punctuation characters other than (< and >), in addition to the normal splitting scheme (splitting into white pages) [3] [4].

Step 4: Building dataset pipeline for training

Using Tensor flow, images with related titles are generated. Dataset objects are decoded and vectorized before being auto-tuned and shuffled if mandatory. There are two steps in the pipeline:

1. Fetch the image from the dataset.
2. Tokenize all five captions related to the image, and then pass it to the list of images and the list of related captions.

Step 5: Model Building

The three models of image captioning model architecture are:

1. CNN model: used to extract characteristics from images using Efficient-Net-B0.
2. Transformer Encoder: The characteristics extracted from the image are transmitted to a Transformer-based encoder, which creates a new representation of the inputs.
3. Transformer Decoder: This model uses the encoder output, as well as the text (sequences) as inputs then, compares both, and selects the best text sequence to use for generating captions.

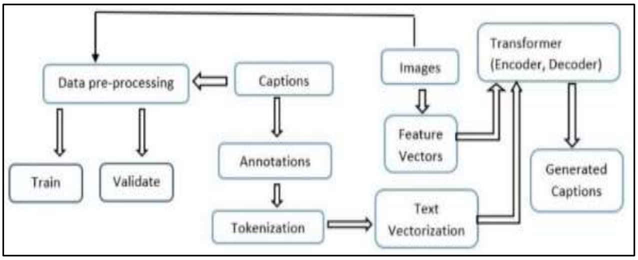


Figure 1. Algorithm of the proposed model.

5 MODEL BUILDING

5.1 Convolution Neural Networks (CNN)

A CNN/ConvNet is a Deep Learning system that assigns importance to an input image (learnable weights, biases), and distinguishes between several aspects/objects in the picture [26]. ConvNet needs remarkably less pre-processing than supplementary classification algorithms. It is capable of handling photos that have been translated, rotated, resized, and altered in perspective. CNN is utilized in NLP [5][6][7][8]. It is more effective than other algorithms since it can predict images with the greatest precision [13].

The dataset is pre-trained using the CNN EfficientNet B0 which was trained with one million photos and more from the ImageNet database. This network can sort images into several categories of one thousand object categories like keyboard, mouse, and several other objects. The size of the input network's picture is 224 × 224 pixels. EfficientNet is a neural network proposed by Google AI. EfficientNet has significantly fewer parameters than the other networks, yet it achieves comparable or even superior results.

5.2 Attention Mechanism

The Transformer's fundamental architecture is the attention mechanism, which was inspired by the attention in the human brain. Attention in transformers is facilitated with the help of query, key, and value [1].

Key: A key is a label of a word and is used to distinguish between different words. Query: Check all available keys and selects the one that matches best. So, it represents an active request for specific information.

Value: Key and values always come in pairs. When a query matches a key, not the key itself but the value of the word gets propagated further. A value is information a word contains.

There are three different attention mechanisms in the Transformer architecture. One is between the encoder and the decoder. This type of attention cross attention when keys and values are created by a separate sequence than queries. If the keys, values, and queries are formed from the same sequence then it is called self-attention. There is one self-attention mechanism in the encoder and one in the decoder which is explained in the next section [15].

5.3 Transformer: Attention-based Encoder and Decoder Architecture

The encoder converts an input sequence into an abstract continuous vector representation that contains all of the input's learned data. The decoder then takes that continuous representation and outputs a single output one step at a time, while also feeding it as the previous output. This happens until the end of the sentence token is generated [2].

Multi-headed Attention: Self-Attention
Multi-head Attention in the encoder uses a self-attention mechanism that allows the model to associate each individual word in the input with other words in the input. The words (input) are fed into three independent fully linked layers to generate a query, key, and value vectors in order to attain self-attention.

The Scaled Dot-Product Attention can be described below in equation (1).

$$\text{Attention}\ (Q, K, V) = softmax(\frac{QK^T}{dk})V \tag{1}$$

Each Query in the Q matrix represents an individual word in the sequence, K (keys) represents all the other words in the Q matrix as vector representations, and V represents the values of all the other words in the Q matrix. Dot product multiplication is used to multiply the queries and keys to generate a score matrix that specifies how much weight a word should be given compared to other words. As a result, each word will be assigned a score that matches the scores of other words in the time step. The higher the score, the more focused the word is. This is how keys and queries are mapped then scores are reduced by √dk times i.e divided by the square root of the dimension of the keys. The weights on the values are calculated using the Softmax function. As a result, higher scores are elevated and lower scores are dropped, helping the model to gain confidence in which words to focus on. The attention weights are then multiplied by the (V) value vector to produce an output vector, which is then passed into the linear layer to be processed. Before applying self-attention, the query, key, and values are separated into end vectors to make it a multi-headed attention competition. Each Head generates an output vector, which is then combined into a single vector before passing through the final linear layer, which learns something new, giving the encoder model additional representation power [14].

Decoder Multi-head attention 1:
Look ahead Mask:
A look-ahead mask is used to restrict the decoder from looking at upcoming to- kens. It is applied before the softmax calculation and after the scores have been scaled. The mask is a zeroed and negative infinities-filled matrix of the same size as that of the scaled scores matrix. When this mask is applied to Scaled scores, a matrix of scores with the top right triangle filled with negative infinities is achieved. As a result, the output of the first multi-headed attention is now a masked output vector having information on how the model should attend to de-coder inputs [19].
Decoder Multi-head attention 2:
The first Multi-head Attention Layers' output and encoders' output (keys and queries) are fed as input to this layer [16]. Then both are compared, letting the decoder determine which encoder's input to concentrate on. For further processing, the result of the second multi-headed attention passes through a point-wise feed-forward layer [23] [24] [25].
Linear Classifier:
The point-wise feed-forward layer's output is passed to a final linear layer, which then accesses a classifier. The classifier is the largest number of classes you have; for instance, if you have 1,000 classes for 1,000 words, the classifier's output size will be 1,000. The classifier's output is passed back into the softmax layer. For each class, the softmax layer generates probability scores ranging from 0 to 1. The highest probability scores' index is used, which equals our expected word. The decoder then adds the output to the list of decoded input and repeats the process until the <end> token is predicted. The decoder generates the output in this manner. It can be stacked n layers high, with each layer receiving input from the encoder and the layer before it. The model can learn to identify and focus on different combinations of attention from its attention head by stacking layers, potentially increasing its prediction capacity

6 TRAINING THE MODEL

6.1 Time taken and parameter calculation during training

Google Collab is used for training the model. And accuracy is checked at various epoch values like (10, 20, 30) it required 4 hours to complete the entire training process. At the end, the model weights are saved which can be used for evaluating next time without training the model again.

6.2 Evaluation Metric

BLEU (Bilingual Evaluation Understudy Score):
The BLEU algorithm can be used to analyze the similarity of machine-translated text. It is used to evaluate the produced caption's quality [17] [18]. The score for a perfect match is 1, whereas the score for a perfect mismatch is 0. The score was established for assessing automated machine translation system predictions [20][21][22]. There are five convincing advantages:

- Calculation is both inexpensive and quick.
- It is simple to understand.
- It is language-agnostic (Independent), and has a profound relationship with human evaluation.
- It enjoys widespread acceptance.

Determining BELU Score:
The predicted caption is "The weather is nice." References:
1. The sky is clear 2. The weather is really good
Every token is 1-gram or unigram, and each word pair is a bigram comparison, therefore the approach works by calculating exact n-grams in prediction translation to n- grams in the original text. The comparison is conducted regardless of word order [11]. To begin, the unigram/bigrams are made out of the predicted caption and references using equation (2).

$$Modifiedngramprecision = \frac{\text{Max number of times ngram occurs in reference}}{\text{Total number of ngrams in hypothesis}} \tag{2}$$

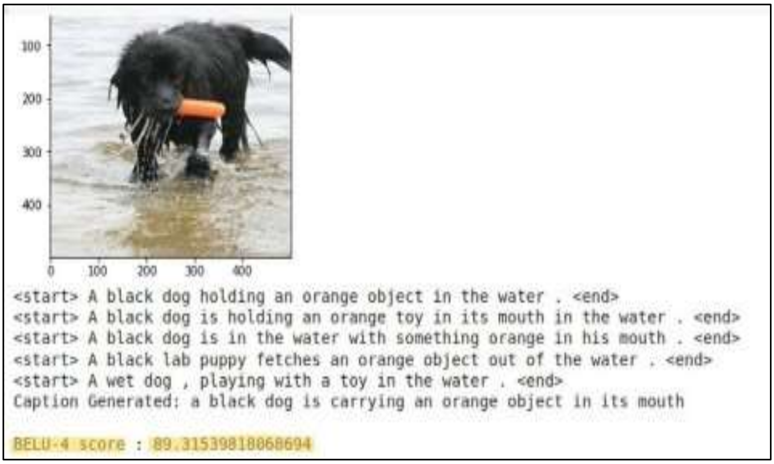
In comparison to the specified 5 reference captions, BLEU indicates how good the predicted caption is.

7 TESTING THE MODEL AND RESULTS

The first step is to upload an image to the program, which may come from the dataset compiled or uploaded by the user. The model tests and prints the relevant description for the user's input, and plays the audio of the generated caption after it has been generated.

Experimental Results:

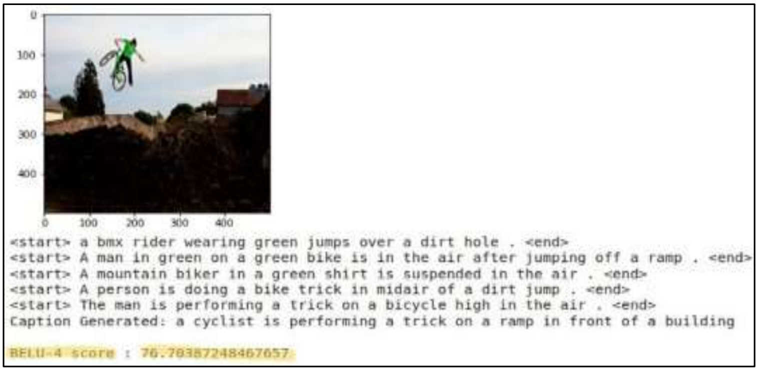
Following are the results and their BLEU scores respectively that have been obtained from the model. These results are classified into two parts, section 'A' images are untrained images from the same Flicker 8k dataset which have some common features as trained images, and section 'B' images are from Internet sources which may be different than trained images, this helps in evaluating the model for external images. Fig. 2 and Fig. 3 are images from untrained dataset. Fig. 4 and Fig. 5 are images from internet source.



```
<start> A black dog holding an orange object in the water . <end>
<start> A black dog is holding an orange toy in its mouth in the water . <end>
<start> A black dog is in the water with something orange in his mouth . <end>
<start> A black lab puppy fetches an orange object out of the water . <end>
<start> A wet dog , playing with a toy in the water . <end>
Caption Generated: a black dog is carrying an orange object in its mouth

BELU-4 score : 89.31539818068694
```

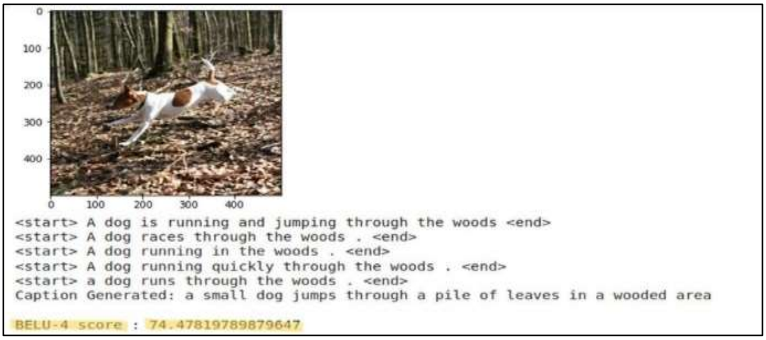
Figure 2. Image from untrained dataset.



```
<start> a bmx rider wearing green jumps over a dirt hole . <end>
<start> A man in green on a green bike is in the air after jumping off a ramp . <end>
<start> A mountain biker in a green shirt is suspended in the air . <end>
<start> A person is doing a bike trick in midair of a dirt jump . <end>
<start> The man is performing a trick on a bicycle high in the air . <end>
Caption Generated: a cyclist is performing a trick on a ramp in front of a building

BELU-4 score : 76.70387248467657
```

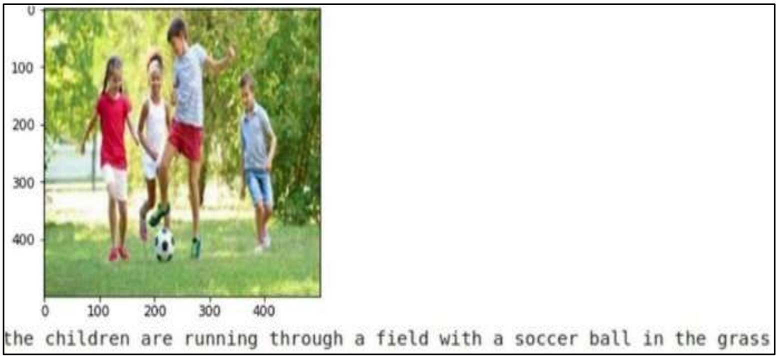
Figure 3. Image from untrained dataset.



```
<start> A dog is running and jumping through the woods <end>
<start> A dog races through the woods . <end>
<start> A dog running in the woods . <end>
<start> A dog running quickly through the woods . <end>
<start> a dog runs through the woods . <end>
Caption Generated: a small dog jumps through a pile of leaves in a wooded area

BELU-4 score : 74.47819789879647
```

Figure 4. Image from internet source.



```
the children are running through a field with a soccer ball in the grass
```

Figure 5. Image from internet source.

8 CONCLUSION AND FUTURE WORK

This paper proposes an attention-based encoder and decoder image caption generator. The attention mechanism implemented after the CNN encoder draws the model's attention to nearly all the pertinent information in the input image, allowing the de- coder to generate the caption using only specific portions of the image. This approach increases the quality of captions compared to typical decoder-encoder-based models. The results achieved are promising, and generated captions are understandable. This experiment finally generates descriptions related to Images uploaded, with good accuracy and it is converted to speech. For future work, this work can be modified by using different CNN models for image feature extraction such as Inception V3, and ResNet which may help in comparing and enhancing this model's accuracy. This model can be trained for large Image Dataset, improving new different objects detection, and it can be upgraded as a mobile application that will help the visually impaired.

References

[1] Xinxin Zhu, Lixiang Li, Jing Liu, Haipeng Peng, and Xinxin Niu. "Captioning Transform- er with Stacked Attention Modules". Applied Science Article, 7 May 2018.

[2] Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S. Zemel. "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention". 32nd International Conference on Machine Learning, ICML 2015.

[3] Andrej Karpathy, Li Fei-Fei. "Deep Visual- Semantic Alignments for Generating Image Descriptions". IEEE Transactions on Pattern Analysis and Machine Intelligence, (Volume: 39, Issue: 4, April 1, 2017).

[4] Haoran Wang, Yue Zhang, and Xiaosheng Yu. "An Overview of Image Caption Generation Methods". Hindawi Computational Intelligence and Neuroscience Volume 2020, Arti- cle ID 3062706.

[5] Zelin Deng, Bo Zhou, Pei He, Jianfeng Huang, Osama Alfarraj and Amr Tolba. "A Position-Aware Transformer for Image Captioning". Computers, Materials and Continua Article, 16 June 2021.

[6] Jun Yu, Member, Jing Li, Zhou Yu, Qingming Huang. "Multimodal Transformer with Multi-View Visual Representation for Image Captioning". JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015.

[7] Shuang Liu, Liang Bai, Yanli Hu and Haoran Wang. "Image Captioning Based on Deep Neural Networks". MATEC Web of Conferences 232, 01052 (2018) EITCE 2018

[8] Simao Herdade, Armin Kappeler, Kofi Boakye, Joao Soares. "Image Captioning: Trans- forming Objects into Words". 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada

[9] Lakshminarasimhan Srinivasan, Dinesh Sreekanthan, Amutha A.L. "Image Captioning - A Deep Learning Approach". International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, NO. 9

[10] Murk Chohan, Adil Khan, Muhammad Saleem Mahar, Saif Hassan, Abdul Ghafoor, Mehmood Khan. “Image Captioning using Deep Learning: A Systematic Literature Review”. (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 11, No. 5, 2020

[11] Chaoyang Wang, Ziwei Zhou and Liang Xu. “An Integrative Review of Image Captioning Research”. ISCME Journal of Physics Conference, 1748(4):042060 January 2021.

[12] Kumari K, Anitha Mouneeshwari, C. Udhaya, R. Jasmitha, R. “Automated Image Caption- ing for Flickr8K Dataset”. International Conference on Artificial Intelligence, Smart Grid and Smart City Applications (pp.679-687), March 2020.

[13] R. Chauhan, K. K. Ghanshala and R. C. Joshi, “Convolutional Neural Network (CNN) for Image Detection and Recognition”. First International Conference on Secure Cyber Computing and Communication (ICSCCC), 2018

[14] Wang, Yiyu and Xu, Jungang and Sun, Yingfei. “End-to-End Transformer Based Model for Image Captioning”. Association for the Advancement of Artificial Intelligence, 2022

[15] V. Agrawal, S. Dhekane, N. Tuniya, and V. Vyas. “Image Caption Generator Using Atten- tion Mechanism”. 12th International Conference on Computing Communication and Net- working Technologies (ICCCNT), 2021 20

[16] H. Parikh, H. Sawant, B. Parmar, R. Shah, S. Chapaneri and D. Jayaswal. “Encoder- Decoder Architecture for Image Caption Generation”. 3rd International Conference on Communication System, Computing and IT Applications (CSCITA), 2020

[17] Cui, Yin and Yang, Guandao and Veit, Andreas and Huang, Xun and Belongie, Serge. “Learning to Evaluate Image Captioning”. CoRR, 2018

[18] Kishore Papineni et al. “BLEU: a method for automatic evaluation of machine translation”. Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics. 2002

[19] Girish Kulkarni et al. “Baby talk: Understanding and generating image descriptions”. Proceedings of the 24th CVPR. Citeseer. 2011.

[20]Micah Hodosh, Peter Young, and Julia Hockenmaier. “Framing image description as a ranking task: Data, models and evaluation metrics”. Journal of Artificial Intelligence Re- search 47 (2013).

[21] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. “Bottom-up and top-down attention for image captioning” and via. arXiv preprint arXiv:1707.07998 (2017).

[22]Rennie, Steven J., et al. “Self-critical sequence training for image captioning”. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017.

[23]Chen, Long, et al. "Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[24]Yang, Xu, et al. “Auto-encoding scene graphs for image captioning”. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019.

[25]Z. Lei, C. Zhou, S. Chen, Y. Huang, and X. Liu, “A Sparse Transformer-Based Approach for Image Captioning”. IEEE Access, vol. 8, pp. 213437-213446, 2020, DOI: 10.1109/ACCESS.2020.3024639.

[26]P Patil, P Narayankar, D Mulimani, M Patil, “Using microscopic images to predict plant diseases in a deep learning environment”, ICT for Competitive Strategies, 807-813.