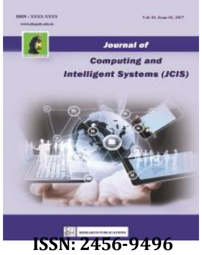
SACRED HEART RESEARCH PUBLICATIONS

Journal of Computing and Intelligent Systems

Journal homepage: www.shcpub.edu.in



ISSN: 2456-9496

GENERATION OF IMAGE CAPTIONS USING DEEP LEARNING TECHNIQUES

Poovizhi M*, Prabhu

Received on 28th MAY 2020, Accepted on 09th JUL 2020

Abstract — *Image Captioning is the concept of developing a description of a specified image. It makes the description of an image by identifying the attributes, objects, and the relationship of the image. It generates the image description in two ways by using the syntactic and semantic methodologies. Deep Learning is a learning domain where it used to generate the captions of the image. In this research, the captions of the image are generated using the deep learning architectures of both CNN (Convolutional Neural Network) and RNN (Recurrent Neural Network). It generates the captions with the aid of the LSTM (Long Short Term Memory) model of deep learning.*

.***Key Words***: Convolutional Neural Network (CNN), RNN (Recurrent Neural Network), Support Vector Machines(SVM)

I. INTRODUCTION

Right Deep learning is a domain where it enables to create the captions of the chosen images. Deep Learning provides various packages to perform the challenges of image. Captioning via Keras, Keras Application, Keras pre-processing, Tensor flow, and long-short Term Memory. In this research, the focus is on the model of VGG16 (visual Geometry Graphics) to generate captions of the images via calculating the image pixels, image specification and image input size, and so on. Deep learning provides the architectures of image captioning like RNN and CNN. With the help of these two architectures, it can know the description of the image. Here, the image is processed through the architectures and VGG16 model. The particular chosen image is converted into convolutional neural network architectures and recurrent neural network architectures followed by the creation of the caption of the image with the long short term memory.

It can be broadly classified into two types, namely

(1). Traditional machine learning-based approach.

(2). Deep machine learning-based approach.

1. TRADITIONAL MACHINE LEARNING-BASED APPROACH.

The appropriate features are extracted from the input images. The input images are moved to processes such as SVM which makes them classify the objects of the images. Noticeably, it is difficult to understand the description of the real-world images such as medical images.

2. DEEP MACHINE LEARNING-BASED APPROACH.

It is used to extract the features of the image through VGG_model architecture. In these techniques, both image and video can able to describe the contents of both things. CNN is widely used for image classification and a set of features from the images and followed by RNN to make the description of the image. **See Fig, 1.1 Flow of DL.**

2. PROBLEM DEFINITION

The exponential rise of technologies leads to the growth of various sources of images that are communicated over the internet or cloud. These data are useful to get an insight into some decision-making purposes for firms and related other institutions. The data can also be virtually present in cloud computing. Since most of the volume of the data using a multimedia form. It is vital to get knowledge of the multimedia data to make the optimal decision making.

There have been several approaches propose to achieve the image caption. To achieve the optimal, the method has to be automated and computationally low cost and there should not be any overhead. Considering this as a motivation in this research the image caption is identified using a novel approach that comprises deep learning techniques. Usually, the image does not have a description yet humans can interpret some knowledge but cannot get a detailed description. Hence there requires a semantic and syntactic approach which automatically incorporates deep learning technique to generate captions of an image.

Corresponding author: E-mail: poomca12@gmail.com,

[1], *Asst. Professor in Computer Science, PG Department of Computer Science, Sacred Heart College (Autonomous),Tiurpattur TamilNadu,605601, India*

[2]*Master of Computer Science, PG Department of Computer Science, Sacred Heart College (Autonomous),Tiurpattur TamilNadu,605601, India*

3. RELATED WORK

Chetan Amritkar et al discussed CNN and RNN architecture and the LSTM model for generating captions of the chosen image. Statistics dataset of Flicker8k datasets were used for training and generating caption of the image [1].

Cheng Wang et al implemented the Bi-directional LSTM Model for captioning the images of trained. They used an effective dataset of flickr30k, flicker8k, MSCOCO on three datasets to evaluate the model functionality high and implemented the multi-task learning concept of this paper [2].

Fang Fang et al was proposed the word-level attention based methodology approached for image captioning. Line level bi-directional approach for extracting the image features and tested with MSCOCO dataset for experimental with word-level attention based [3].

Yan Sun et al have discussed the image captioning technique involved in the cognition task of computing in IOT application areas. The VGG net structure model which provides the image accuracy. The model which was trained to improve the efficiency of accuracy and connected in IoT application areas [4].

Min Yang et al have proposed the multi-purpose knowledge of language image captioning generation via the cross-domain dual algorithms. Evaluated the effectiveness of MLADIC in cross-domain captioning via the MOSCOCO dataset and other source datasets were used for Flicker30K datasets [5].

Parth Shah et al introduced the deep neural architecture model to process the images. NLP plays a major role in this technique of both image processing and also with the help of natural language. By using object detection to capture the image feature extraction to evaluate the standard matrices of the trained model [6].

Philip Blandford et al discussed automatic image captioning is popularity in the field of AI. Normally, a human can understand the image in various ways. Likewise, a machine can be trained to understand the model and extract the features of the images and highlighted the captions in the natural language text processing field. There are different ways of filters of noise removal and their effect can be several properties of the flicker data were analyzed and the performance of the model weight is differentiated. Each model can be weighted on the system memory of the loaded data. Some of the images have sentimental analyzed images on flicker [7].

Harga et al have proposed the technique of deep learning which plays a vital-role or backbone or evidence of the image captioning in the past a few years. Encoder and decoder architectures were used to encode the image and find the similarities of the images that are used to decode the caption of the images. As the form of new features of datasets, that has the sentimental sensing things or description words in different languages. That combines the working of deep learning neural encoder and decoder architectures [8].

Lun Huang et al processed the model of generating the caption of images by provides the consisting of two cascaded agents. The first one has achieved the training process for the number -task performance and number-stage trained to strengthen of which provides the two cascaded agents.

In experimental used the validity MSCOCO dataset to secure the process and classification of the image processing. The sequencer model pre-trained the architectures of encoder and decoder [9].

Jiahe Shi et al proposed the method based on multiple feature-based learning the encoder and decoder progress of image captioning. Likewise, the encoder encodes the structure of an image with weighted layers and decodes gave the prediction according to the base on a model trained. The R-CNN based frameworks were supported to do the progress of the images. MS COCO dataset was tested with higher priorities of the accuracy of the image captioning [10].

Yang Xian et al were discussed the self-guiding based LSTM model where doesn't have a valid dataset for the image training models. It was proposed to handle the real-world sentence when it's unbalanced. Hence, the LSTM model provides the extraction of the images. Experimental was tested with the real world sentence case to predicate the valid output for both syntactically and semantically valid [11].

Xinyu Xiao et al discussed the Deep Hierarchical Encoder and Decoder Neural network architecture for the image caption. We can say that its backbone of image captioning when we choose deep learning is the encoder and decoder associated with the LSTM model to generate different ways of output for single images. Here, the qualitative analysis indicates our model translated the image into a sentence and provides the visualization plotting images of the evolution of the hidden layer states from the different procedural steps of LSTM. For this kind of experimental process both the benchmark and statistics of dataset like Flicker8k and Flicker30k and MSCOCO datasets [12].

4. EXPERIMENTAL SETUP:

Content	Machine Learning	Deep learning
Training Dataset	Small	Large
Choose your features	Yes	No
Available	Many	Few
Training Time	Short	Long

Table 1. Comparison of ML & DL

4.1 PROCESS OF IMAGE CAPTIONING:

There are roles of image captioning to describe the sentences of an image.

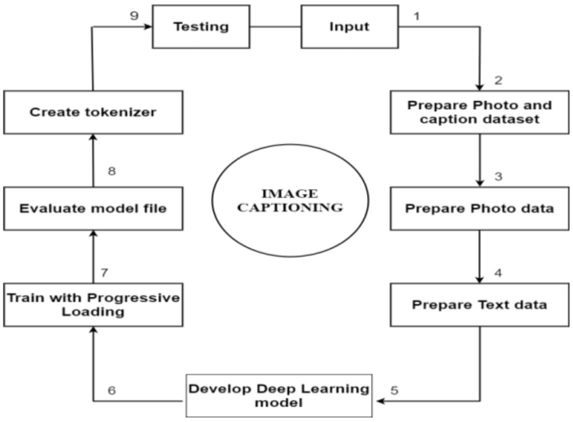


Fig 2. Structure of image captioning

4.2 ARCHITECTURE OF CNN:

It is a kind of deep neural network that was designed from the biological driven models so what researches found how a mammal or how a human perceives an image into the brain in a different layer. So, the convolutional neural network is designed for image classification and hence it has proven very efficient for all the image processing techniques.

CNN has set of filters that are applied to a given input image for processing the number of filters and with that processed filters activates the data of the given image goes to another layer which is called a pooling layer. The pooling layer is a kind of nonlinear downsampling the given image has to process to the next level of understanding. Again the same process goes like a bunch of activation filters which is a convolutional neural network.

The image gets more information from the previous layer which is a process to the next stage to get a gain of the image process. Hence, layers called fully connected. The term says that the fully connected layer which provides every node connected in the next part of node coefficients. It is a heavy data-driven with a load of coefficients to support each node.
We get more output of the single image from the pooling layer. Each of the images gets the top three top five best captions of the image. The convolutional layer gets a single image with multiple captions of pooling layer considerations. Let's consider the input image is 3-dimensional width and height as the input has the number of kernels or filters which have several activation filters that are going to generate for our given image. One filter extracting the color of the image and another filter extract the edges nodes of the images and so on. When we see a single image it consists of non-linearity function, the thing can be recognized as a different non-linearity function but the convolution layer by defined as a linear function so we need to add some non-linearity function to the linearity function of the image, to understand the images. The pooling layer is mainly responsible for reducing the size of the images and coefficients are connected to the edges to give the output of the image.

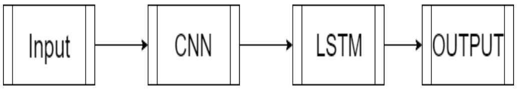
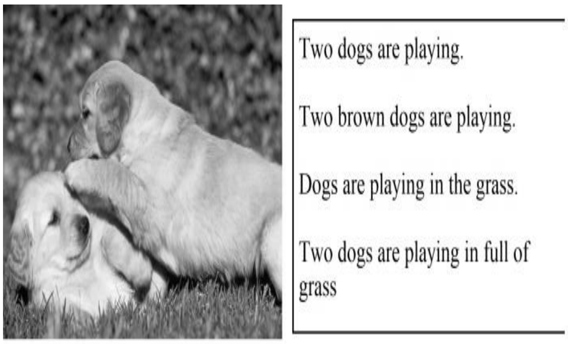


Fig 3. Process of CNN

4.3 VGG16 MODEL:

It was developed by simonyon and Zisserman. VGG stands for Visual Geometry Graphs. It consists of 16 layers it is very fixed appealing and uniform structured. It was fundamentally used for extracting a feature from the images. In VGG16, layer 16 depicts a possible weight layer in the model. It is a convolutional neural network model which contains 3/3 convolutional layers. Mostly, VGGNet provides the 138 million parameters and offers a bit challenging to progress this architecture. Keras providing a vgg16 model to extract the features of images and save them into the file. **See Fig 3.Arcitecture of understanding VGG. as mentioned**



Two dogs are playing.

Two brown dogs are playing.

Dogs are playing in the grass.

Two dogs are playing in full of grass

Fig 4 .output of VGG

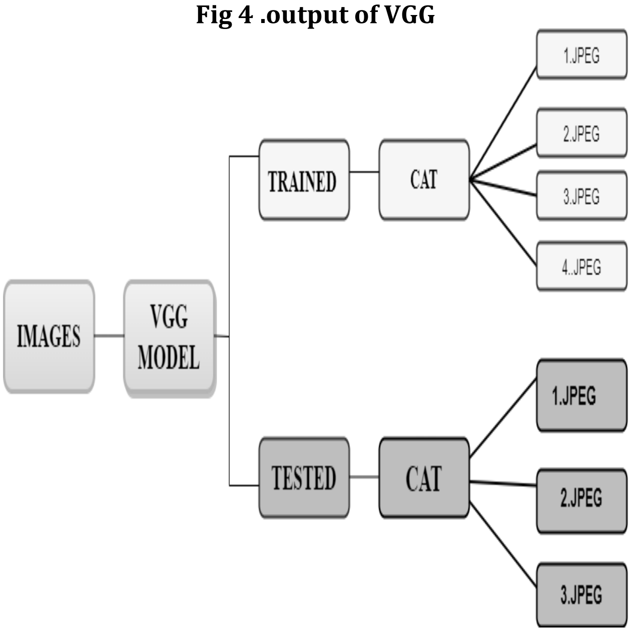


Fig 5. How the image can flow in the VGG process

5. AREAS OF APPLICATIONS:

5.1 DATASET:

The meaningful dataset that is Flicker8K Dataset and Filcker8K Text data were used as input which is available online. User has to request through a form to access the flicker. You can send a request form to access the Flicker Dataset then a notification will be sent to the given email.
The following link can request for a dataset
https://forms.illinois.edu/sec/1713398

- Flicker8k Dataset
- Flicker8k Text

In the Flicker8k dataset, there are 8092 images for image training. From the 8092 images, the train images are 6000 and 1000 images for testing. Each image consists of five different captions that provide descriptions of entities in the image.

In Flicker8k text, there are tokens, annotations, train images, test images.

5.2 ROLES OF CAPTIONING:

1. Make Photo data
2. Make Text data
3. Produce deep learning model
4. Train with progressive loading
5. Estimate the Model file
6. Create Tokenizer
7. Generate Captions

The following things are the roles of image captioning. From the following things, the data can be prepared, trained, evaluated, and tested and captions generated.

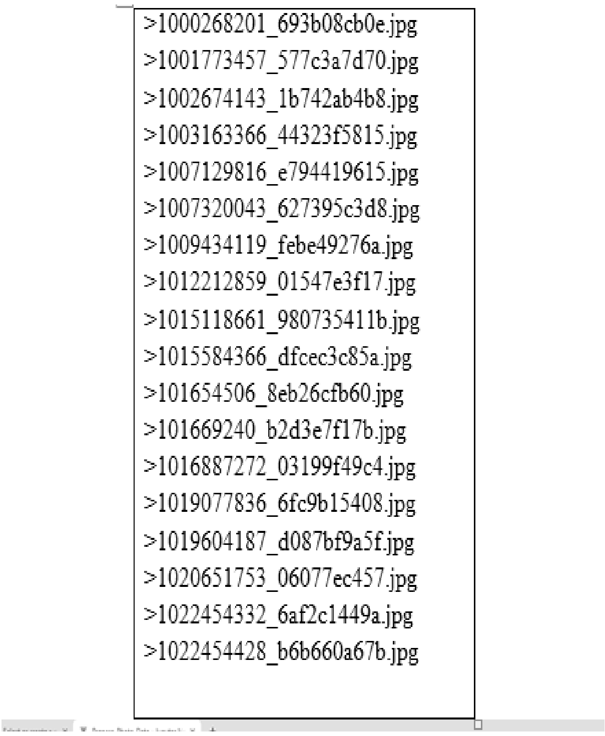
1. Make Photo data:

- Keras provides the pre-trained model directly to extract the functions of the images.
- Depend upon your internet connection speed the model can be extracted.
- It may take a few minutes to depend on the size of your system memory capacity.
- By loading the VGG model in the Keras using the VGG structure.
- By removing the last layer from the loaded model to use to predicate the classification for an image.
- Keras provides a tool for remodeling the loaded photos into the process of preferred size for the model extraction. (e.g. 3 channel 224 x 224 pixels of the image.)
- The file loaded and started to extract the images from the loaded images and finally saved it into features.pkl file extension.

2. Make Text Data:

- The text file contains descriptions and token of the images. Each image consists of five descriptions.
- Each photo has a different unique id for extraction of the images, while loaded and prepared for text data.
- The file size of loaded images is 8092. The size of the vocabulary is 8763. These files finally saved into descriptions .txt file format.
- The text data prepares the tokens of the trained images.
- The text data helps to understand the five different captions of the single trained images.
- Each of the trained images can have five different captions, according to the model

Trained data, it generates the single caption for that chosen image.



```
>1000268201_693b08cb0e.jpg
>1001773457_577c3a7d70.jpg
>1002674143_1b742ab4b8.jpg
>1003163366_44323f5815.jpg
>1007129816_e794419615.jpg
>1007320043_627395c3d8.jpg
>1009434119_febe49276a.jpg
>1012212859_01547e3f17.jpg
>1015118661_980735411b.jpg
>1015584366_dfcec3c85a.jpg
>101654506_8eb26cfb60.jpg
>101669240_b2d3e7f17b.jpg
>1016887272_03199f49c4.jpg
>1019077836_6fc9b15408.jpg
>1019604187_d087bf9a5f.jpg
>1020651753_06077ec457.jpg
>1022454332_6af2c1449a.jpg
>1022454428_b6b660a67b.jpg
```

Fig 6. Extraction of images

Loaded: 8092
Vocabulary size: 8763

Table 2. The output size of text data

The loaded and vocabulary size is mentioned above in the table.

1000268201_693b08cb0e.jpg#0	A child in a pink dress is climbing up a set of stairs in an entryway.
1000268201_693b08cb0e.jpg#1	A girl going into a wooden building.
1000268201_693b08cb0e.jpg#2	A little girl climbing into a wooden playhouse.
1000268201_693b08cb0e.jpg#3	A little girl climbing the stairs to her playhouse.
1000268201_693b08cb0e.jpg#4	A little girl in a pink dress going into a wooden cabin.

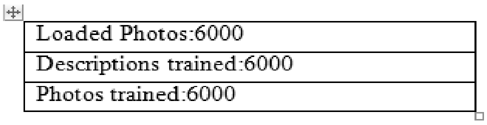
Table 3. Multi-descriptions of a single image

3. PRODUCE DEEP LEARNING MODEL:

- Processing the model
- Describe the model
- Fixing the model

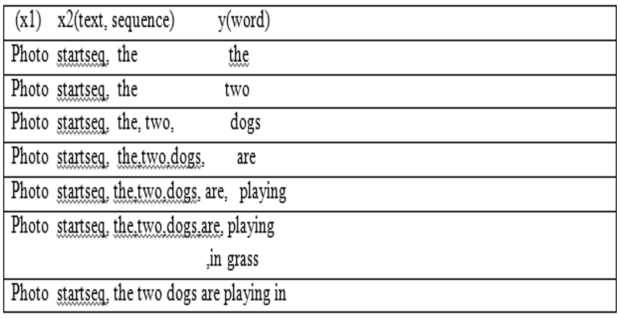
PROCESSING THE MODEL

The idea is to process the photo data and text data which helps in describing the model. With the help of those data provided the list of images and also with descriptions, the development dataset finally stored as each model file.



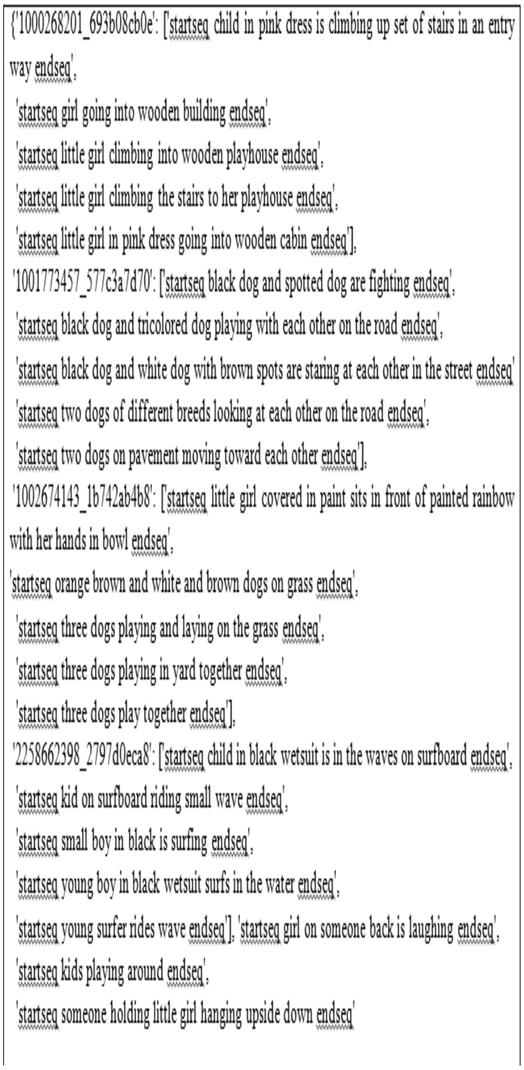
Loaded Photos:6000
Descriptions trained:6000
Photos trained:6000

Fig 7. Defined the volume of trained data's



(x1)	x2(text, sequence)	y(word)
Photo	startseq, the	the
Photo	startseq, the	two
Photo	startseq, the, two,	dogs
Photo	startseq, the,two,dogs,	are
Photo	startseq, the,two,dogs, are,	playing
Photo	startseq, the,two,dogs,are, playing	,in grass
Photo	startseq, the two dogs are playing in	

Fig 8. Caption process



{'1000268201_693b08cb0e': ['startseq child in pink dress is climbing up set of stairs in an entry way endseq',
'startseq girl going into wooden building endseq',
'startseq little girl climbing into wooden playhouse endseq',
'startseq little girl climbing the stairs to her playhouse endseq',
'startseq little girl in pink dress going into wooden cabin endseq'],
'1001773457_577c3a7d70': ['startseq black dog and spotted dog are fighting endseq',
'startseq black dog and tricolored dog playing with each other on the road endseq',
'startseq black dog and white dog with brown spots are staring at each other in the street endseq'
'startseq two dogs of different breeds looking at each other on the road endseq',
'startseq two dogs on pavement moving toward each other endseq'],
'1002674143_1b742ab4b8': ['startseq little girl covered in paint sits in front of painted rainbow with her hands in bowl endseq',
'startseq orange brown and white and brown dogs on grass endseq',
'startseq three dogs playing and laying on the grass endseq',
'startseq three dogs playing in yard together endseq',
'startseq three dogs play together endseq'],
'2258662398_2797d0eca8': ['startseq child in black wetsuit is in the waves on surfboard endseq',
'startseq kid on surfboard riding small wave endseq',
'startseq small boy in black is surfing endseq',
'startseq young boy in black wetsuit surfs in the water endseq',
'startseq young surfer rides wave endseq']. 'startseq girl on someone back is laughing endseq',
'startseq kids playing around endseq',
'startseq someone holding little girl hanging upside down endseq'

Fig 9. Image contains descriptions

The following structure shows the defining and fitting of the layers.

Totally 20 epoch layer generates the model File with the size of images is 6000 and it should be trained with value loss and value accuracy. Each model saves model_1.h5 for evaluating the testing file process it generates captions. Each epoch model has some weight according to the memory size

STRUCTURE OF PLOT:

By plotting the structure of the network that better helps to understand the two streams input.

See fig. Structure of understanding of two streams input process as mentioned at last.

Fixing model:

The model learns fast and quickly by the trained data. The skill of training data and the whole model is sent into the model file.

Finally, the testing process, by saving the model file to the skill of the training dataset.

Loaded dataset :	**8092**
Trained Dataset :	**6000**
Photos Trained :	**6000**
Descriptions Trained :	**6000**
Vocabulary Size :	**7579**
Descriptions size :	**34**
Valid Dataset :	**1000**
Valid Descriptions :	**1000**
Valid : Photos :test :	**1000**
Tested :	**1000**

Table 4. Fitting the model

3. Train with Progressive Loading Model:

The training of the model consumes a lot of system memory. For that, the system capacity needs a minimum memory of 32GB or 64 GB. The training model expects a specific size of RAM for the testing of images.

4. Estimate Model File:

When the model file is fitted. To test the dataset the skill of its predictions on the holdout dataset.

The word started with "startseq" and ended with "endseq. The function named below generate_desc() implements its process of generating a textual description word in the form of Natural Language English.

5. Create Tokenizer:

The Tokenizer created with the file of descriptions. From the trained dataset 6000, created the tokenize file of each image.

Using Tensor Flow backend:

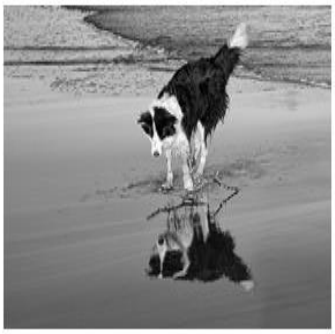
Trained Dataset: 6000

Descriptions –trained : 6000 [tokens]

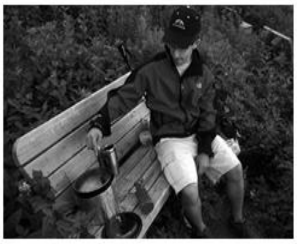
Table5. The output of the tokenizer file

6. Generate captions:

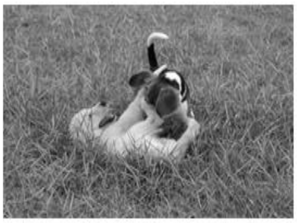
- Almost ready to generate the captions for entire photographs via the model file.
- By providing the tokenizer file for encoding the processed words of the trained model.
- The maximum length of sequences with appropriate captions.
- An image that can be described in the sentence with a length of description size is 34.



startseq dog is running across the beach endseq



startseq man in red shirt is sitting on the sidewalk endseq



startseq dog is running through the grass endseq

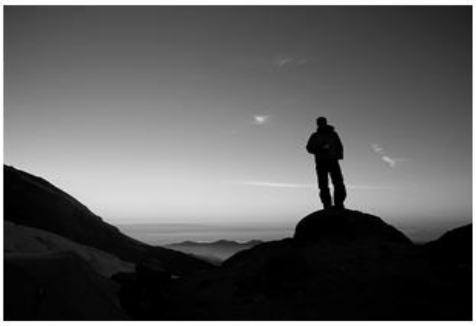
CONCLUSION

The work aims at the presents a model, which is a neural network that can automatically view an image and generate appropriate captions in a natural language like English.

The model is trained to give the sentence or description from a chosen image. The descriptions or captions obtained from the model are categorized into:

- Defined without errors
- Defined with minor errors
- Define somewhat related to image
- Defined unrelated to the image

The higher dataset used for training will yield increased accuracy.



startseq man in red shirt is climbing up rock face endseq

Fig 11.The output of image description

7. RESULTS AND ACCURACY:

A. Datasets

These datasets consist of images and description form of sentences in natural language such as English.

The results are defined in the table.

Dataset	**Loaded**	**Trained**	**Valid**	**Tested**
Flicker	8092	6000	1000	1000

In these datasets, each image is described with 5 different captions that are related to the visible and impartial.

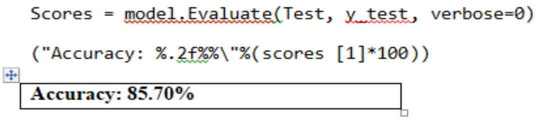
B. Results

The model trained for a layer of 20 epochs.

The number of epochs is large, it helps to lessen the loss accuracy to 3.74.

By considering consider the maximum dataset then process the more epochs for an accurate result

C. Accuracy: #Calculate Accuracy



```
Scores = model.Evaluate(Test, y_test, verbose=0)

("Accuracy: %.2f%%\"%(scores [1]*100))
```

Accuracy: 85.70%

REFERENCES

[1] Chetan Amritkar, Vaishali Jabade, "Image Caption Deep Learning Technique", Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), IEEE Publications, ISSN: 5386-5257, 2018, PP.1-4.

[2] CHENG WANG, HAOJIN YANG, and CHRISTOPH MEINEL, "Image Captioning with Deep Bidirectional STMs and Multi-Task Learning", Multimedia Computing Communication.Appl,Vol 14.2S,Article 40,ACM Trans Publications, 2018,PP.1-20.

[3] Fang Fang, Hanli Wang, Pengjie Tang, "Image Captioning with Word Level Attention", 25th IEEE International Conference on Image Processing (ICIP), IEEE Publications, ISSN: 4799-7061, 2018, PP.1278-1282.

[4] Yan Sun, Lida Xu, Ling Li, Boyi Xu, Changbao Yin, Hongming Cai, "Deep Learning Based Image Cognition Platform for IoT applications", 15th International Conference on e-Business Engineering (ICEBE), IEEE Publications, 2018, ISSN:5386-7992, PP. 9-16.

[5] Min Yang, Wei Zhao, Wei Xu, Yabing Feng, Zhou Zhao, Xiaojun Chen, Kai Lei, "Multitask Learning for Cross-domain Image Captioning", IEEE Transaction on Multimedia, IEEE Publications, ISSN: 1520-9210, PP. 1-15

[6] Parth Shah, Vishvajit Bakrola, Supriya Pati, "Image Captioning using Deep Neural Architectures", International Conference on Innovations in information Embedded and Communication Systems (ICIIECS), IEEEPublications,2017, ISSN: 5090-3294.

[7]PhilippBlandfort,TusharKarayil,DamianBorth,AndreasDengel, "ImageCaptioningintheWild:HowPeopleCaptionImagesonFlickr"MountainView,CA,USA,ACMPublications,ISBN:4503-55092017, PP.21-29

[8]I. Hrga, M.Ivašić-Ko, "Deep Image CaptioningAnOverview"inMIPRO, PP.995-1000.

[9]. Lun Huang, Wenmin Wang∗, Gang Wang, "Image captioning with two cascaded agents" IEEE Publications, 2019, ISSN: 5386-4658, PP.4110-4114.

[10] Jiahe Shi, Yali Li, Shengjin Wang, "Cascade attention: Multiple features based learning For image captioning",2019, IEEE Publications, ISSN: 5386-6249, PP. 1970-1974.

[11] Yang Xian, Member, IEEE, Yingli Tian, Fellow, IEEE, "Self-Guiding Multimodal LSTM - when we do not have a perfect training dataset for image captioning", IEEE Publications, 2019, PP. 1-13.

[12] Xinyu Xiao, Lingfeng Wang, Kun Ding, Shiming Xiang, Chunhong Pan, "Deep Hierarchical Encoder-Decoder Network for Image Captioning", IEEE Publications, 2018, PP.1-16

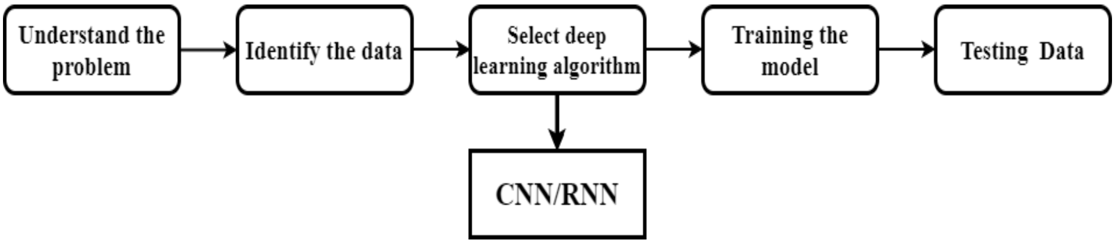
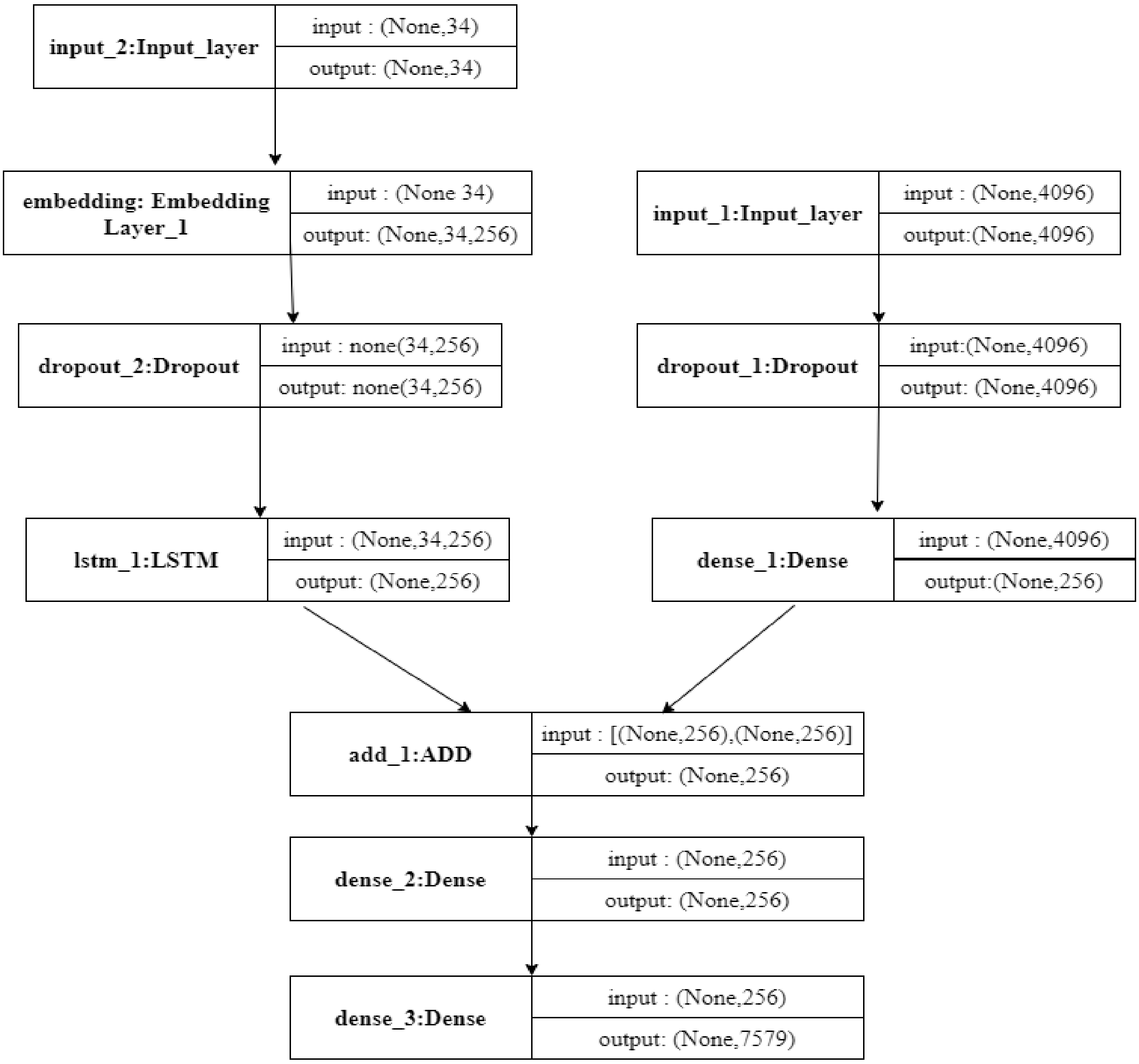


Fig 1.1 Flow of DL



Plot Caption of the Deep learning model

Fig 5.1 Structure of plot