International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 06 | June - 2021 ISSN: 2582-3930

# Image Captioning Web App

**Gaurav Joshi[1], Dr. Amita Goel[2], Ms. Vasudha Bahl [3] , Ms. Nidhi Sengar [4]**

[1] B-tech scholar, Department of IT Maharaja Agrasen Institute of Technology

[2] Professor, Department of IT Maharaja Agrasen Institute of Technology

[3] Assistant Professor, Department of IT Maharaja Agrasen Institute of Technology

[4] Assistant Professor, Department of IT Maharaja Agrasen Institute of Technology

---

**Abstract -** Deep Learning is a relatively new field that has gotten a lot of attention since it can recognize objects with greater precision than ever before. NLP is another field that has had a significant impact on our lives. It indicates the effect of NLP that it has progressed from providing a readable summary of the texts to analyzing mental disorders. The problem of image captioning involves NLP and Deep Learning. Image captioning can be used to describe photographs in a meaningful way. Describing an image entails more than merely recognizing things; in order to effectively describe an image, we must first identify the things included in the image, followed by the relationship between those items. We used a CNN-LSTM framework in this research. We used a CNN-LSTM framework in this research. CNN will be used to extract visual attributes, while LSTM will be used to try to construct relevant words. This research also looks at how Image captioning is used and the significant issues that come with it.

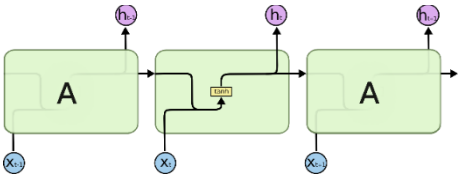
***Key Words***: Image captioning, Webapp, Deep Learning

## 1.INTRODUCTION

To Simply put, image captions are an automatic image description generator that can help users to automatically generate a description of the presented image. The project model aims to get the input image and generate a sentence description of the basic content of the image. Describing the content of an image in a simple and easy to understand language is one of the complex and basic tasks. With the help of advanced technology and the availability of data sets, model building has become a possible task.

With the help of sight vision, humans can accurately define and describe the description of any image that comes their way. Like humans, computers are also developing at a rapid pace, they can recognize the basic actions of classifying objects and recognize their state and characteristics. However, defining images precisely in simple, clear language that humans can understand has become a relatively new and challenging task. Automatic image captioning performs its function in a number of tasks. The first step in understanding an image begins with the extraction of the image and its related environment, that is, the objects are "books" and "tables". In the next stage, the relationship between the detected objects has been identified for further evaluation, that is, for book and table objects, the relationship between the two is defined as "book on the table"

Once the objects and their relationships are defined, further evaluation will be carried out in the text description. The sequence of words must be located in some way so that it makes sense once formed and justifies the actual relationship of the objects placed in the image. For the first task, which is to extract features from an image, we used a convolutional neural network (CNN) in this project. It should be noted that "extract features" in most cases refers to removing the last softmax layer. For the second part, the generation of text descriptions, we will use short-term memory (LSTM). LSTMs are a special type of RNN , which is used to avoid the long-term dependency problem that often occurs in the case of RNN.



## 2. RELATED WORK

The work [1] clearly shows that deep learning has received a lot of attention in recent years, and much progress has also been made in this field. This is also obvious when we look at the statistics. Only 4 successful articles were published in 2015, but the popularity of this field has grown exponentially since then, as can be seen from the 57 articles published in 2017-2018.

Elamri [3] also proposed a solution based solely on the CNN-LSTM architecture. The model uses CNN to extract characteristics from a given image and then feeds them into the RNN or LSTM model. Later, the RNN or LSTM model describes the image in a grammatically correct way and can describe what happens in the image. The paper also discussed the advantages of the image caption model for the visually impaired. To help visually impaired people in society, if properly developed, image captions can become a useful device. This project considers all studies that have been conducted in the field in the past and is also affected by these studies. Most of the works we studied use CNN and RNN-based architectures. An interesting finding from previous research on this topic is that "adding more layers to the model does not necessarily mean that we will get higher accuracy."

Work by Di Lu and Spencer Whitehead [2] suggests that a new task can be created, and an image description of the task will be provided as input to the system. The document also mentions that the current use of Image Captioning lacks specific motivation for the entities that constitute the basic structure of the image. In this article they also proposed a solution to this problem. The article suggests that the CNN-LSTM model should be trained to be able to generate titles based on the rendered image.

CONVOLUTIONAL NEURAL NETWORKS ARCHITECTURES

| Architecture | Top-1 Accuracy | Top-5 Accuracy | Year |
|---|---|---|---|
| Alexnet | 57.1 | 80.2 | 2012 |
| Inception-V1 | 69.8 | 89.3 | 2013 |
| VGG | 70.5 | 91.2 | 2013 |
| Resnet-50 | 75.2 | 93 | 2015 |
| InceptionV3 | 78.8 | 94.4 | 2016 |

**Table -1:** Sample Table format

## 3. Methodology AND IMPLEMENTATION

As already discussed in the abstract, the basic goal of the project is to provide subtitles for images in real time. The dataset used to build this project is the Flickr8k dataset. In the Flickr8k data set, each picture has 5 corresponding titles. The data set provides 6,000 images for training purposes, 1,000 images for verification purposes, and the remaining 1,000 images for testing purposes. The project has been divided into five main tasks:

### Data Cleaning

Get the image id from the data set and create a dictionary to map the image with the title. The token.txt file takes image identification and subtitles as input. From this token.txt file, we will only map each image with its own subtitles. The total words in our data set are close to 37,000. Now we have to reduce these words, because this will affect our calculations, and if a word arrives in a shorter time, then there is no point in using it. Now we have set the threshold to 10, so if the frequency of a word is lower than 10, we don't consider it. After filtering the words according to the threshold frequency, we only have 1,845 words, which constitutes our vocabulary dictionary.

### Image encoding

We can now use photos as input to our model, but unlike humans, machines cannot understand images when viewing them. So we need to convert the photo into a code so that the machine can recognize the pattern. For this, I used the transfer study, . We used a pre-reviewed version trained on a large data set. We extracted the functions of these patterns and applied them to our photos. For this research, I used a version of Resnet50 that has been trained on Imagenet. We can easily import this model from keras. Program module.

```
In [117]: start = time()


          for ix,img_id in enumerate(train):
              img_path = IMG_PATH+"/"+img_id+".jpg"
              encoding_train[img_id] = encode_image(img_path)

              if ix%50==0:
                  print("Encoding in Progress Time step %d "%ix)

          end_t = time()
          print("Total Time Taken :",end_t-start)
```

**Vocabulary Segmentation/ Tokenization**

In this step, we need to segment all vocabulary words. Alternatively, we can use a tokenizer in Keras to accomplish this task.

**Defining the model**

To outline the shape of our version, we will use the keras model in the functional api. It has three main steps:

Process collections of text content

Extract feature vectors from photos

Interpreting the output with the aid of concatenating the above layers.

**Creating Web Application**

We have used Flask, Flask is a micro web framework written in Python. Along with flask we have used standard web technologies like HTML,CSS and Bootstrap, While HTML is used to provide structure, bootstrap and CSS is used for styling the web application.

The web app provides a feature using which user can upload any image to the application and after that our app will try to describe the image and will predict the caption.

Creating a form so that user can input the image and on this image we will make the prediction

```
<div class="container">

    <div class="jumbotron">
        <div class="container">
            <h1 class="heading"> Image Captioning</h1>
            <p> Upload your image to generate a caption... </p>
        </div>
    </div>



    <form class="form-inline" method="POST" action="/" enctype="multipart/form-data">
        <label><b> Image :    </b></label>
        <input class="form-control" type="file" name="userfile" placeholder="Your image">
        <button type="submit" class="btn btn-primary"> Submit</button>
    </form>

    <br>
    <br>
```

```
import flask
from flask import render_template, redirect, request

import Caption_it

app = flask.Flask(__name__)

@app.route('/')
def hello():
    return render_template("index.html")

@app.route('/', methods= ['POST'])
def marks():
    if request.method == 'POST':
        f = request.files['userfile']
        path = "./static/{}".format(f.filename)
        f.save(path)

        caption = Caption_it.caption_this_image(path)

        result_dic = {
        'image' : path,
        'caption' : caption
        }

    return render_template("index.html", your_result =result_dic)


if __name__ == '__main__':
    app.run(debug = True)
```

## 4. DISCUSSION

### 4.1 Challenges faced

#### 4.1.1 Detecting Multiple Objects

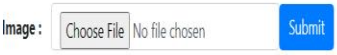
Although we have the ability to detect multiple objects in machine learning models available to us today but models can't always interpret the relationships between those objects. Therefore, the model can't always give an accurate description of the image. In addition, the data set that we use is only 8k images for the FlickR8K data set. If you want to explain the model correctly, and grammatically, you must train too many data sets in too large format in the correct format. When talking about large data sets, large data sets also takes a great time to train, in addition to this testing remains a very important problem that we must have to deal.

#### 4.1.2 Availability of Datasets

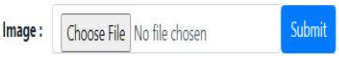
The most common datasets commonly used for image captions are Flickr8k, Flickr30K, and MS-COCO. Most of these data sets are now in English. As mentioned in literature survey, so far we have many datasets that can be used to train our model, but most of the training samples are in English or Chinese. This is a very important topic. If we want to use image caption templates in practical applications, we need the availability of training samples in multiple languages.

### 4.2 Applications

To help visually impaired people in society, if properly developed, image captions can become a useful device. It can be a difficult task to develop an automatic image captioning system that can provide an accurate description of the image as a stand-alone system. Here, the captured image can be used as input for automatic image captions. Therefore, loud noise can be used to provide output, which can help the visually impaired to better understand the surrounding environment.





### 5. Conclusion

Deep learning can bring significant changes to society, and image captions have made significant progress in recent years. Image captions can provide a large number of applications in various fields, such as agriculture and intelligent system monitoring. The surprising thing is that image captions are not used in fields such as traffic analysis, and traffic analysis can benefit a lot from it. This research builds on several previous articles and research in the field. The research looked for several specific models and strategies for image captions. We found CNN to extract features and content is the most suitable model and is also widely used. For generating descriptions, frequently used models are RNN and LSTM (a special type of RNN).

### References

[1] S. ALBAWI and T. A. MOHAMMED, "Understanding of a Convolutional Neural Network,"in ICET, 2017.
[2] S. Hochreiter, "LONG SHORT-TERM MEMORY," Neural Computation, December 1997.
[3] O. Vinyals, A. Toshev, S. Bengio and D. Erhan,"A Neural Image Caption Generator," CVPR 2015 Open Access Repository, vol. Xiv, 17 November 2014.
[4] D. S. Whitehead, L. Huang, H. and S.-F. Chang, "Entityaware Image Caption Generation,"in Empirical Methods in Natural Language Processing, 2018.
[5] C. Elamri and T. Planque, "Automated Neural Image Caption Generator for Visually Impaired People," California, 2016.
[6] Xu, Kelvin, et al. "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention" Computer Science,2048-2057,2015.
[7] Papineni, K. "BLEU: a method for automatic evaluation of MT" 2001.
Shuang Liu, Liang Bai, Yanli Hu and Haoran Wang “Image captioning based on deep neural networks”.
[8] Ranzato, Marc'Aurelio, et al. "Sequence Level Training with Recurrent Neural Networks." *Computer Science* (2015)

[9] Mao, Junhua, et al. "Explain Images with Multimodal Recurrent Neural Networks." *Computer Science* (2014)

[10] Sundermeyer, M., et al. "Comparison of feedforward and recurrent neural network language models." *IEEE International Conference on Acoustics, Speech and Signal Processing IEEE*, 8430-8434. (2013)

[11] Simonyan, Karen, and A. Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." *Computer Science* (2014)

[12] Szegedy, Christian, et al. "Going deeper with convolutions." *IEEE Conference on Computer Vision and Pattern Recognition IEEE*, 1-9. (2015).

[13] Mao, Junhua, et al. "Explain Images with Multimodal Recurrent Neural Networks." *Computer Science* (2014).

[14] Aneja, Jyoti, A. Deshpande, and A. Schwing. "Convolutional Image Captioning." (2017)

[15] Jeel Sukhadiya, Harsh Pandya, Vedant Singh Comparison of Image Captioning Methods