IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 7, May 2022
Impact Factor: 6.252

Review of Case Study of Image Caption Generator

Abhijeet A. Karande, Mahesh A. Raut, Swapnil R. Mukare, Samir B. Shaikh, Prof. G. G. Patil
Department of Computer Science Engineering
SVERI's College of Engineering, Pandharpur, India

Abstract: We are very interested in how machines can automatically describe the content of images using human language. In order to gain a deeper insight of this computer vision topic, we decided to implement current state-of-the-art image caption generator Show, attend and tell: Image caption generator with visual attention Our neural network-based image caption generator is implemented in Python powered by Pytorch machine learning library. We have identified three major components in our pipeline: (1) data prepossessing; (2) Convolutional Neural Network (CNN) as an encoder; (3) attention mechanism; We evenly distributed the three components described above among our group and each member has made equal contributions to push the project forward. We have successfully finished the implementation of the all three components. Our implementation of this image caption generator has achieved a very decent accuracy.

Keywords: Image Caption Generator , Machine learning, CNN

I. INTRODUCTION

Training computers to be able to automatically generate descriptive captions for images is currently a very hot topic in Computer Vision and Machine Learning. This task is a combination of image scene understanding, feature extraction, and translation of visual representations into natural languages. This project shows some great promises such as building assistive technologies for visually impaired people and help automating caption tasks on the internet. There are a series of relevant research papers attempting to accomplish this task in last decades, but they face various problems such as grammar problems, cognitive absurdity and content irrelevance.

1.1 Objectives

The main objective of the project on Image Caption Generator is to create or generate a caption for the provided image. It manages to scan all the image and generate the caption by capturing all the details in image (like people, animals, colours, etc.). This project is totally built of server level and thus anyone can use it. The purpose of this project is to built a model that generates a caption for image easily. It track all the details and according to it creates a suitable caption. You saw an image and your brain can easily tell what the image is about, but can a computer tell what the image is representing? Computer vision researchers worked on this a lot and they considered it impossible until now! With the advancement in Deep learning techniques, availability of huge datasets and computer power, we can build models that can generate captions for an image

Benefits of Image Caption Generator:

Automation is procedure of generating a suitable caption for the provided image. To overcome the defects of the existing, manual exam system, automation was introduced by the computerization of organization we get many benefits
The general significance of the project can be defined as follows
1. To avoids wastage of time for finding caption.
2. To minimizing efforts of user.
3. To give speed and time save service for user.
4. To find the caption easily.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 7, May 2022
Impact Factor: 6.252

II. LITERATURE SURVEY

There are many methods which are used for image paragraph captioning. These methods aim to generate simple sentences for an image.

Methods For Sentence Generator:

Sequence Level Training with Neural Network: It generates a syntactically and semantically correct sequence of consecutive words to form a sentence. Many natural language processing applications use language models to generate text. These models are typically trained to predict the next word in a sequence, given the previous words and some context such as an image. However, at test time the model is expected to generate the entire sequence from scratch. This discrepancy makes generation brittle, as errors may accumulate along the way. We address this issue by proposing a novel sequence level training algorithm that directly optimizes the metric used at test time, such as BLEU or ROUGE. On three different tasks, our approach outperforms several strong baselines for greedy generation. The method is also competitive when these baselines employ beam search, while being several times faster. Lstm , Data As Demonstrated to end back propagation are used here.

Bottom up and top down for image captioning and visual answering question: Problems combining image and language understanding such as image captioning and visual question answering continue to inspire considerable research at the boundary of computer vision and natural language processing. In both these tasks it is often necessary to perform some fine-grained visual processing, or even multiple steps of reasoning to generate high quality outputs. These mechanisms improve performance by learning to focus on the regions of the image that are salient and are currently based on deep neural network architectures. Top-down attention mechanism, Faster R-CNN Lstm are used to achieve this

III. MAIN FUNCTIONS

Building the Python based Project

Let's start by initializing the jupyter notebook server by typing jupyter lab in the console of your project folder.

It will open up the interactive Python notebook where you can run your code. Create a Python3 notebook and name it.

2.training_caption_generator

First, we import all the necessary packages

Getting and performing data cleaning

We will define 5 functions

1. load_doc(filename)
2. all_img_captions(filename)
3. cleaning text(descriptions)
4. text vocabulary (descriptions)
5. save descriptions(descriptions, filename)

3. Loading dataset for Training the mode

For loading the training dataset, we need more functions:

• load_photos(filename) – This will load the text file in a string and will return the list of image names.

• load_clean_descriptions(filename, photos) – This function will create a dictionary that contains captions for each photo from the list of photos. We also append the and identifier for each caption. We need this so that our LSTM model can identify the starting and ending of the caption.

• load_features(photos) – This function will give us the dictionary for image names and their feature vector which we have previously extracted from the Xception model.

4.Training the model

To train the model, we will be using the 6000 training images by generating the input and output sequences in batches and fitting them to the model using model.fit_generator() method. We also save the model to our models folder. This will take some time depending on your system capability.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 7, May 2022
Impact Factor: 6.252

5.Testing the model

The model has been trained, now, we will make a separate file testing_caption_generator.py which will load the model and generate predictions. The predictions contain the max length of index values so we will use the same tokenizer.p pickle file to get the words from their index values.

REFERENCES

[1]. https://machinelearningmastery.com/
[2]. https://scholar.google.com/
[3]. https://www.wikipedia.org/