Software Requirements Specification

For

Image Caption Generator

Prepared by

Specialization	SAP ID	Name
B.Tech CSE AIML	500075359	Niharika Agrawal
B.Tech CSE AIML	500076519	Prabhraj Singh
B.Tech CSE AIML	500075307	Pradumn Nathawat

Department of Informatics
School Of Computer Science
UNIVERSITY OF PETROLEUM & ENERGY STUDIES,
DEHRADUN- 248007. Uttarakhand

Table of Contents

Revision History

Date	Change	Reason for Changes	Mentor Signature

1. INTRODUCTION

1.1 Purpose of The Project

Automatically generating captions of an image is a task very close to the heart of scene understanding — one of the primary goals of computer vision. Not only must caption generation models be powerful enough to solve the computer vision challenges of determining which objects are in an image, but they must also be capable of capturing and expressing their relationships in a natural language. In the last few years, it has become a topic with growing interest in machine learning and the advances in this field lead to models that (depending on which evaluation) can score even higher than humans do. Nevertheless, image captioning is a very complex task as it goes beyond the sole classification of objects in pictures. The relation between the objects and the attributes has to be recognized. Finally, this information must be expressed in a natural language like English.

1.2 Target Beneficiary

Image captioning can for instance help visually impaired people to grasp what is happening in a picture. Furthermore, it could enhance the image search of search engines, it could simplify SEO by automatically generating descriptions for the pictures or improve online marketing and customer segmentation by identifying customer interests through interpreting their shared images via social media platforms.

1.3 Project Scope

This project aims at generating captions for images using neural language models. There has been a substantial increase in the number of proposed models for image captioning tasks since neural language models and convolutional neural networks(CNN) became popular. Our project has its base on one of such works, which uses a variant of Recurrent neural network coupled with a CNN. We intend to enhance this model by making subtle changes to the architecture and using phrases as elementary units instead of words, which may lead to better semantic and syntactic captions.

1.4 References & GitHub link

https://www.irjet.net/archives/V7/i4/IRJET-V7I41167.pdf
https://data-flair.training/blogs/python-based-project-image-caption-generator-cnn/
https://datascience.stackexchange.com/questions/26947/why-do-we-need-to-add-start-s-end-s-symbols-when-using-recurrent-neural-n
https://www.hindawi.com/journals/cin/2020/3062706/
https://fairyonice.github.io/Develop_an_image_captioning_deep_learning_model_using_Flickr_8K_data.html#:~:text=Download%20the%20Flickr8K%20Dataset&text=The%20images%20were%20chosen%20from,by%20submitting%20the%20request%20form.
https://www.hindawi.com/journals/cin/2020/3062706/

https://github.com/prabhrajsingh/MINOR-2-Image-Caption-Generator

2. PROJECT DESCRIPTION

2.1 Reference Model

CNN is crucial in working with images. It takes as input an image, assigns importance (weights and biases) to various aspects/objects in the image, and differentiates one from the other. The CNN makes use of filters(also known as Kernels) which help in feature learning(detect abstract concepts, like Blurring, Edge Detection, Sharpening, etc), much the same as a human brain identifying objects in time and space.

Long Short Term Memory networks – usually just called "LSTMs" – are a special kind of RNN, capable of learning long-term dependencies. Remembering information for long periods is practically their default behavior, and this behavior is controlled with the help of "gates". While RNNs process single data points, LSTMs can process entire sequences. Not only that, they can learn which point in the data holds importance, and which can be thrown away. Hence, the only relevant information is passed on to the next layer.

2.2 Libraries

- **tensorflow:**
  TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications.

- **keras:**
  Keras is an API designed for human beings, not machines. Keras follows best practices for reducing cognitive load: it offers consistent & simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear & actionable error messages.

- **numpy:**
  NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object and tools for working with these arrays. It is the fundamental package for scientific computing with Python. It contains various features.

- **matplotlib:**
  Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram, etc.

- **OpenCV:**
  OpenCV is an image processing package for Python language. It incorporates lightweight image processing tools that aid in editing, creating, and saving images. It supports a large number of image file formats including BMP, PNG, JPEG, and TIFF. The library encourages adding support for newer formats in the library by creating new file decoders.

2.3 Characteristics of Data

Flilckr8K contains 8,000 images that are each paired with five different captions which provide clear descriptions of the salient entities and events. The images were chosen from six different Flickr groups, and tend not to contain any well-known people or locations, but were manually selected to depict a variety of scenes and situations.

2.4 SWOT Analysis

Strength -

The Flickr8k dataset is a relatively small dataset that can be applied easier and efficiently on low-end systems. The use of transfer learning helps the model generate more accurate data. We don't have to do everything on our own, we use the pre-trained model that has been already trained on large datasets and extract the features from these models and use them for our tasks.

Weakness -

Since sentences should make sense and are one of the most common ways to communicate, the computational generation of texts recognizable as a sentence that makes sense and is within the context of the image is difficult.

Opportunities -

Image captioning can for instance help visually impaired people to grasp what is happening in a picture. Furthermore, it could enhance the image search of search engines, it could simplify SEO by automatically generating descriptions for the pictures or improve online marketing and customer segmentation by identifying customer interests through interpreting their shared images via social media platforms.

Threats -

The model needs a START and END for the model to stop generating tokens

2.5 Project Features

- Our project's main feature is the use of CNN and LSTM to generate tokens for the image (input). Those tokens are the descriptive text that tries to explain the image as a human brain does.

- The use of the inception model provides us with the feature vectors on its own using a method called transfer learning where the inception model is previously trained on an imagenet dataset that had 1000 images to classify.

- With the use of the Attention model, we are improving the relevance of the caption (description) generation as the attention model will look for the attention weights of the neighboring features and also consider the previously predicted words as an input as well

- In the output cell we will also have a feature to make the text as a speech feature for the visually impaired people for whom the project was initially intended for.

2.6 User Classes and Characteristics

It can be used for the following user classes:

- Image Editing Applications: The image captioning model automates and accelerates the closed captioning process for digital content production, editing, delivery, and archival.

- Social Media Posts: For social media, artificial intelligence is moving from discussion rooms to underlying mechanisms for identifying and describing terabytes of media files. It enables community administrators to monitor interactions and analysts to formulate business strategies.

- For visually impaired persons: The advent of machine learning solutions like image captioning is a boon for visually impaired people who are unable to comprehend visuals.

2.7 Design and Implementation Constraints

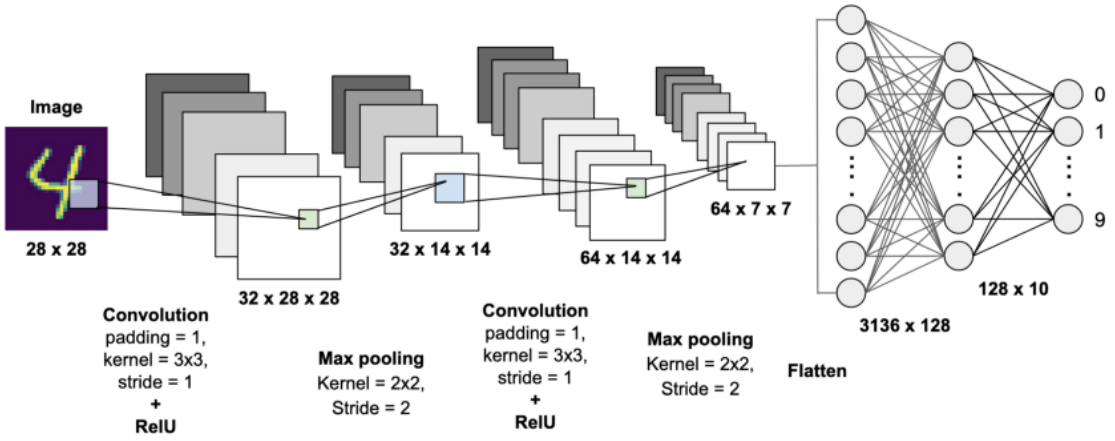
- Cannot use large datasets such as flickr30k or Microsoft COCO dataset due to the high-end system requirements.

- They do not make intuitive feature observations on objects or actions in the image, nor do they give an end-to-end mature general model to solve this problem.

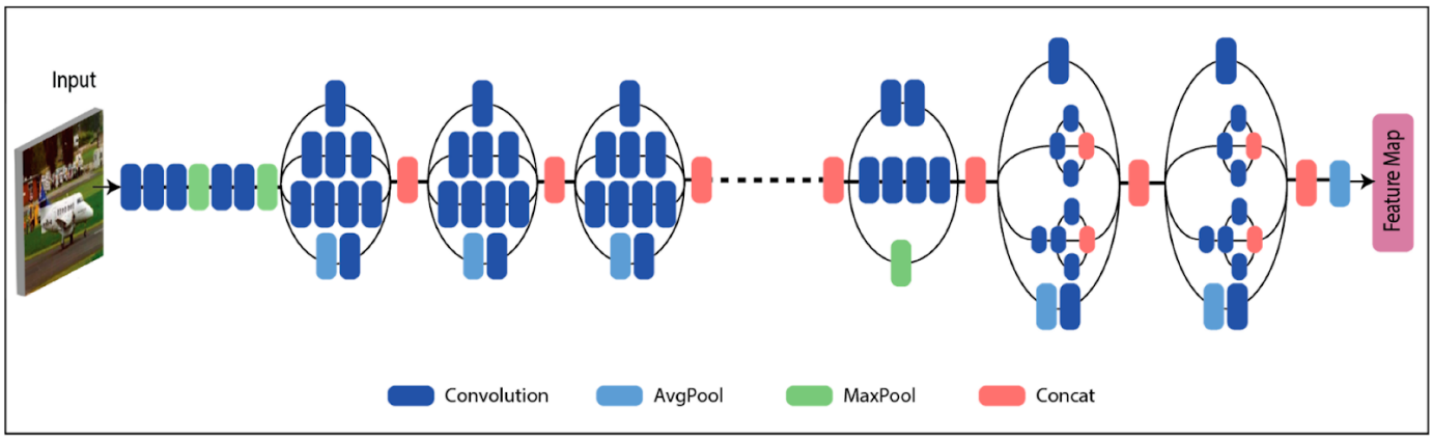
- The model may overfit quickly because of the small data.

2.8 Design diagram

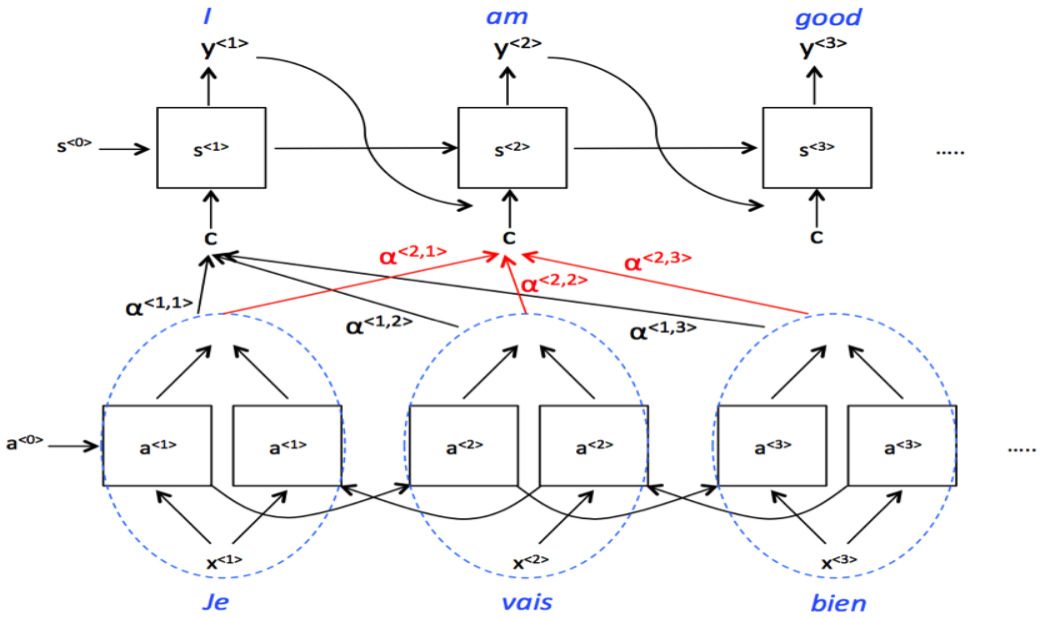
A general CNN diagram



For our Image Caption model, we need only the image feature maps, and do not need the Classifier i.e., the last layer.



Attention helped the model focus on the most relevant portion of the image as it generates each caption word.



3. SYSTEM REQUIREMENTS

3.1 User Interface:

The user interface will be a simple GUI created within the python notebook itself where we simply provide an image and will have the caption presented to us.

3.2 Software Interface:

- OS: Windows 7 and above, Recommended: Windows 10.
- CPU: Intel processor with 64-bit support
- Disk Storage: 8GB of free disk space.
- google colab notebook in Python

3.3 Dataset Interface

Our Flickr8k Dataset comprises 8000(8k) images of the .jpg extension and can be found online.

3.4 Protocols

- The project is done in the python programming language and needs to follow the rules and syntax of the python language.
- During the implementation of the CNN model, we have to extract the feature vector from the image but due to transfer learning inclusion, we need to make some changes and then integrate it with our model.
- For the attention model, we will be directly using those features with some previous outputs for the model to know about the context

4. NON - FUNCTIONAL REQUIREMENTS

4.1 Performance requirements

With the use of the google colab notebook we will be taking the help of its GPU acceleration feature where a complex procedure can be performed drastically fast and with the google drive linked directly to the notebook we need not manually upload the dataset.

5. CONCLUSION

When given an image, the caption (description) of that particular image is being generated as an output product

Although image caption can be applied to image retrieval, video caption, video movement and a variety of image caption systems are available today, experimental results show that this task can still have better performance systems and improvement.

It mainly faces the following three challenges:

first, how to generate complete natural language sentences like a human being.

second, how to make the generated sentence grammatically correct.

third, how to make the caption semantics as clear as possible and consistent with the given image content

5.1 Output

```
Ttest_image = pred_caption_audio(len(image_test), True, weights = (0.5, 0.25, 0, 0))
Image.open(Ttest_image)
```

```
/usr/local/lib/python3.7/dist-packages/nltk/translate/bleu_score.py:490: UserWarning:
Corpus/Sentence contains 0 counts of 2-gram overlaps.
BLEU scores might be undesirable; use SmoothingFunction().
  warnings.warn(_msg)


 BLEU score: 50.0


Real Caption: people gather on bridge near UNK UNK tree


Prediction Caption: some people are crossing green bridge over river
```