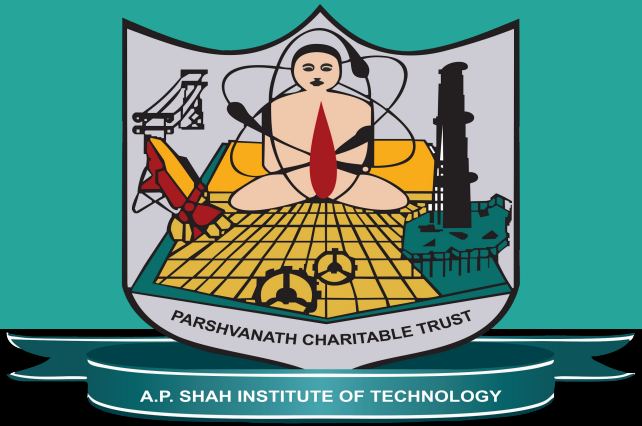


Computer Engineering Department

A.P. Shah Institute of Technology

G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615

UNIVERSITY OF MUMBAI

Academic Year 2019-2020

Image Captioning Using Deep Neural Networks

Members:
Mrunal S Jadhav(16102030)
Aditya G Joshi(16102017)

Project Guide:
Prof.Sachin Malave

TABLE OF CONTENTS

Introduction to Image Captioning

Introduction

- Image captioning is a task that a machine learns to generate natural language sentences to describe the salient parts of an image.
- A description must capture not only the objects contained in an image, but it also must express how these objects relate to each other as well as their attributes and the activities they are involved in.
- Thus, accurate image captioning is a challenging task that requires advancing the state of the art of both computer vision and natural language processing.

CAN YOU WRITE A CAPTION?

PROJECT CONCEPT AND INITIATION

1.1 Abstract

- The visual world is populated with a vast number of objects, the most appropriate labelling of which is often ambiguous, task specific, or admits multiple equally correct answers.
- A quick glance is sufficient for a human to understand and describe what is happening in the picture. The task is to transform a sentence S written in its source language, into its translation T in the target language, by maximising the probability P(T|S).
- A combination of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) which seeks to progress directly from image features to text can be progressed to define a single end-to-end model to maximize the likelihood of the target description sentence , given an image, instead of requiring sophisticated data preparation or a pipeline of specifically designed models.
- Thus we can develop a generative model, a probabilistic framework, based on deep recurrent architecture that combines advances in computer vision and machine translation to generate natural sentences describing an image.

1.2 Objectives

- Image captioning is a task that a machine learns to generate natural language sentences to describe the salient parts of an image. Being able to automatically describe the content of an image using properly formed English sentences is a very challenging task.

- Generating complete and natural image descriptions automatically has large potential effects, such as titles attached to news images, descriptions associated with medical images, text-based image retrieval, information accessed for blind users, human-robot interaction. These applications in image captioning have important theoretical and practical research value.

- The meaningful description generation process of high level image semantics requires not only the understanding of objects or scene recognition in the image, but also the ability to analyse their states, understand the relationship among them and generate a semantically and syntactically correct sentence.

1.3 Literature Review

- The first significant work in solving image captioning tasks was done by Ali Farhadi [4] where three spaces are defined namely the image space, meaning space and the sentence space where mapping is done from the respective image and sentence space to the meaning space.

- O.Vinyals and team, in the work [1], introduced a novel approach of using (CNN) and (RNN) for image captioning tasks. Convolutional neural networks were used to extract features from the images. So, CNN acts as a encoder, first for classification of tasks and the last layer output is provided as the input to (RNN). (RNN) acts as a decoder that generates sentences. LSTM networks (Long Short Term Memory) was the type of RNN used.

1.4 Problem Definition

- In this project we hope to achieve more precise and accurate image captioning model. Here, we propose to follow this elegant recipe, replacing the encoder RNN by a deep convolution neural network (CNN).

- There will be an end-to-end system for the problem. It is a neural net which is fully trainable using stochastic gradient descent. The model combines state-of-art sub-networks for vision and language models.

- Finally, we wish to yields significantly better performance compared to state-of-the-art approaches. we propose a fully trainable attribute-based neural network founded upon the CNN+RNN architecture, that can be applied for image captioning.

1.5 Scope

- Translation work is achieved by using an “encoder” RNN that reads the source sentence and transforms it into a rich fixed-length vector representation, which in turn is used as the hidden state of a “decoder” RNN that generates the target sentence.

- Replacing the encoder RNN by a deep CNN can produce a rich representation of input by embedding it in a fixed-length vector, so that this representation can be used for variety of tasks.

- Developing a single end to end network to have more accurate feature extraction and efficiently generate textual description which can provide detailed information about the given image.

1.6 Technology stack

1. **Colab** - Colaboratory is a Google research project created to help disseminate machine learning education and research
2. **Pytorch** - used for applications such as computer vision and natural language processing.
3. **Numpy** - NumPy is the fundamental package for scientific computing with Python.
4. **Pandas** – It is a software library written for the Python programming language for data manipulation and analysis.
5. **Keras** – It is an Open Source Neural Network library written in Python and a high-level API wrapper for the low-level API that runs on top of Theano or Tensorflow.
6. **Sklearn** - is a free software machine learning library for the Python programming language.
7. **Tensorflow** – It is an end-to-end open source platform for machine learning.

1.7 Benefits for environment & Society

1. Helps visually impaired to understand the image by converting the captioned text into speech.

2. Helps colour blind and other vision problem patients to understand image more effectively.

3. Generate captions for images which can promote safety and protection of environment

4. Determine various pollutants present in a given image and caption them so as to reduce its generation and manage it.

Datasets

Flickr 8K, Flickr 30K, Google’s Conceptual Captioning Dataset

Flickr 8K Dataset

- Data is properly labelled. For each image 5 captions are provided and the dataset is available for free.

- Flickr8k_Dataset: Contains a total of 8092 images in JPEG format with different shapes and sizes. Of which 6000 are used for training, 1000 for test and 1000 for development.

- Flickr8k_text : Contains text files describing train_set ,test_set. Flickr8k.token.txt contains 5 captions for each image i.e. total 40460 captions.

- The size of the training vocabulary is 7268

Flickr 30K Dataset

- Flickr30K Entities, which augments the 158k captions from Flickr30k with 244k coreference chains, linking mentions of the same entities across different captions for the same image, and associating them with 276k manually annotated bounding boxes.

- Such annotations are essential for continued progress in automatic image description and grounded language understanding. They enable us to define a new benchmark for localization of textual entity mentions in an image.

Google's Conceptual Captioning Dataset

- On September 5, 2018 Google introduced Conceptual Captions, a new dataset consisting of ~3.3 million image/caption pairs that are created by automatically extracting and filtering image caption annotations from billions of web pages.

- Furthermore, because images in Conceptual Captions are pulled from across the web, it represents a wider variety of image-caption styles than previous datasets, allowing for better training of image captioning models.

- To generate this dataset, a Flume pipeline processes billions of Internet web pages, extracting, filtering, and processing candidate image and caption pairs, and keeping those that pass through several filters.

- The Training split consists of 3,318,333 image-URL/caption pairs, with a total number of 51,201 total token types in the captions, validation has 28,355 image-URL/caption pairs, with a total number of 13,063 total token types in the captions and testing has 22,530 image-URL/caption pairs, with a total number of 11,731 total token types in the captions.

Dataset Samples:

Flow of Model

Flow of Model

Input Image
(224x224x3)

Pretrained CNN
using ImageNet dataset

CNN

Feature vector
at fc layer
(1x1x2048)

Linear

LSTM

LSTM

LSTM

LSTM

Softmax

Softmax

Softmax

Softmax

<start>

Giraffes

standing

<end>

W_{emb}

W_{emb}

W_{emb}

<start>

Giraffes

other

Convolution and Feature Extraction

Convolution Neural Network

- A Convolutional Neural Network (CNN) is a Deep Learning algorithm which can take in an input image, assign learnable weights and biases to various aspects/objects in the image and be able to differentiate one from the other.

- A ConvNet is able to successfully capture the Spatial and Temporal dependencies in an image through the application of relevant filters.

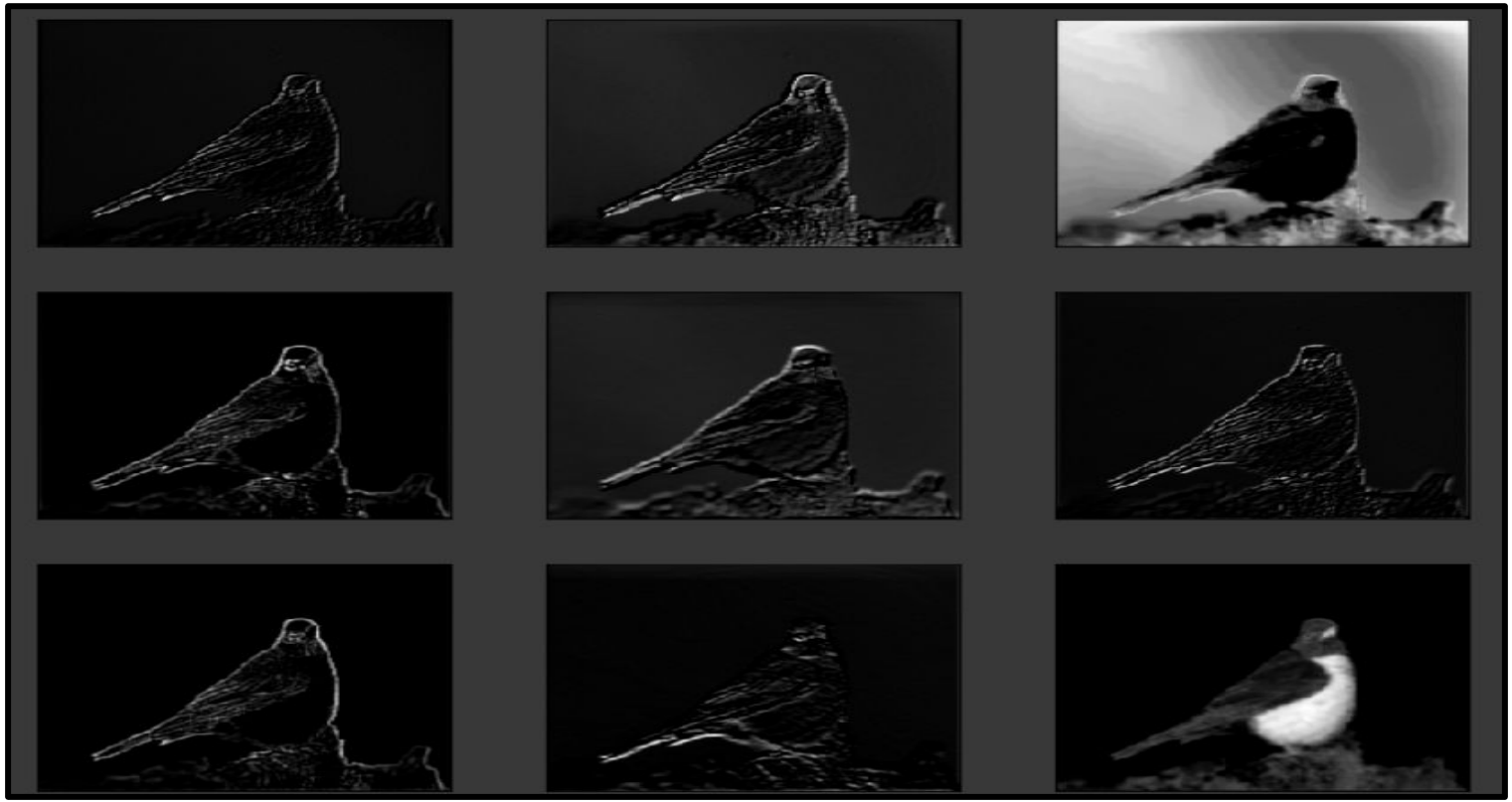
- The role of the ConvNet is to reduce the images into a form which is easier to process, without losing features which are critical for getting a good prediction.

- Using keras in sequential type of model ,we add convolution layers and other layers like max pooling ,activation etc to build our cnn model.

- Using these CNN layers we extract all the features from given image.

- These features are then forwarded as input to LSTM blocks for caption generation.

Convolution Filter Visualization



Text Processing

Text Pre-processing

- The captions need to be preprocessed before using them.
- Model may treat a word which is in the beginning of a sentence with a capital letter different from the same word which appears in the sentence but without any capital letter. Therefore, we change all the words to lowercase.
- We may want the words, but without the punctuation like commas and quotes. We also want to keep contractions together.Python provides a constant called string.punctuation that provides a great list of punctuation characters.
- Removes hanging 's' and other letters earlier specified with apostrophe.
- Since the numbers are not useful in analysis we may remove them.

Text Pre-processing

- Once we have cleaned the data, we can build a vocabulary representing our dataset.
- Text data must be encoded as numbers to be used as input or output for machine learning and deep learning models.
- fit_on_text : Creates a vocabulary based on word frequency
- text_to_sequence Transforms each text into sequences of integers. Takes each word from the text and replaces with corresponding integer value from word_index

Glove Vectors

Glove vectors

- Glove is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space.

- The Euclidean distance (or cosine similarity) between two word vectors provides an effective method for measuring the linguistic or semantic similarity of the corresponding words.

- Sometimes, the nearest neighbors according to this metric reveal rare but relevant words that lie outside an average human's vocabulary.

Glove Vector Representation

```
king - man = queen - woman
france - paris = britain - london
france - paris = italy - rome
paris - france = rome - italy
france - french = england - english
japan - japanese = china - chinese
japan - japanese = italy - italian
japan - japanese = australia - australian
december - november = july - june
miami - florida = houston - texas
einstein - scientist = matisse - painter
china - rice = chinese - bread
man - woman = he - she
man - woman = uncle - aunt
man - woman = brother - sister
man - woman = friend - wife
man - woman = actor - actress
man - woman = father - mother
heir - heiress = queen - princess
nephew - niece = uncle - aunt
france - paris = japan - tokyo
france - paris = china - beijing
february - january = october - november
france - paris = italy - rome
paris - france = rome - italy
```

```
neighbors of: king
        prince
        queen
        ii
        emperor
        son
neighbors of: france
        french
        belgium
        paris
        spain
        netherlands
neighbors of: japan
        japanese
        china
        korea
        tokyo
        taiwan
```

Recurrent Neural Networks

GRU and LSTM

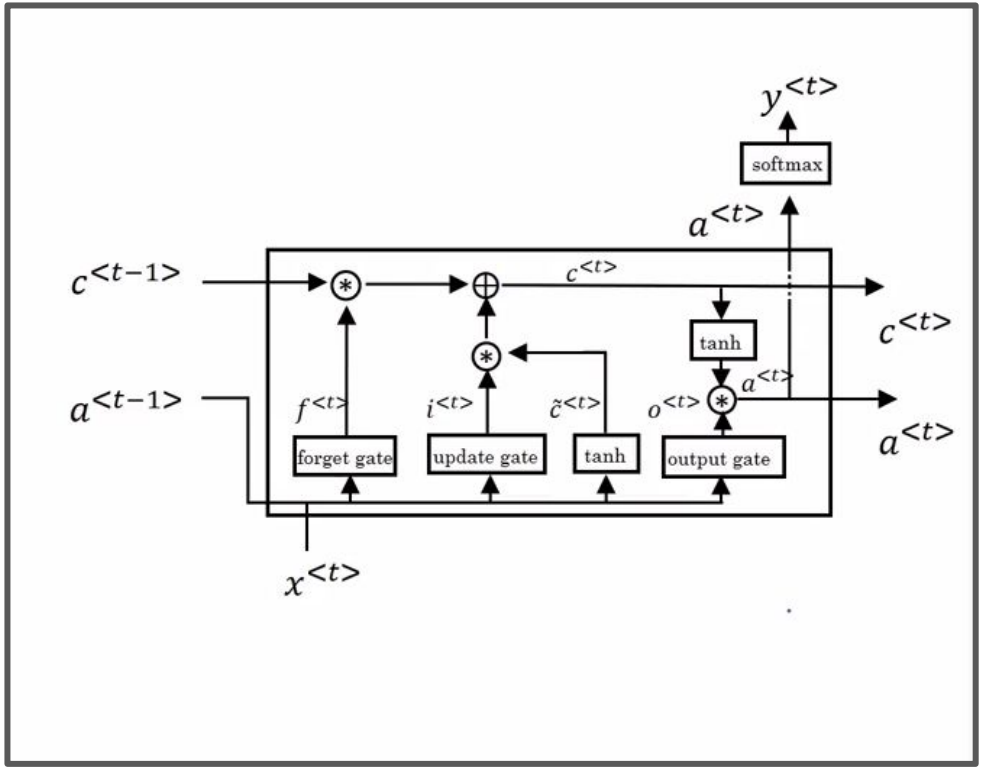
Recurrent Neural Networks

- A recurrent neural network (RNN) is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behavior

- Long short-term memory (LSTM) is an artificial recurrent neural network (RNN) architecture used in the field of deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. It can not only process single data points (such as images), but also entire sequences of data (such as speech or video).

- A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell.

LSTM In A Picture



$$\tilde{c}^{<t>} = \tanh(W_c[a^{<t-1>}, x^{<t>}] + b_c)$$
$$\Gamma_u = \sigma(W_u[\, a^{<t-1>}, x^{<t>}] + b_u)$$
$$\Gamma_f = \sigma(W_f[\, a^{<t-1>}, x^{<t>}] + b_f)$$
$$\Gamma_o = \sigma(W_o[\, a^{<t-1>}, x^{<t>}] + b_o)$$
$$c^{<t>} = \Gamma_u * \tilde{c}^{<t>} + \Gamma_f * c^{<t-1>}$$
$$a^{<t>} = \Gamma_o * \tanh c^{<t>}$$

BLEU Score

BLEU Score

- BLEU, or the Bilingual Evaluation Understudy, is a score for comparing a candidate translation of text to one or more reference translations.

- BLEU is evolved version of max precision.

- NLTK provides the sentence_bleu() function for evaluating a candidate sentence against one or more reference sentences.

- The reference sentences must be provided as a list of sentences where each reference is a list of tokens.

- The candidate sentence is provided as a list of tokens.

BLEU Score

$$p_n = \frac{\sum_{C \in \{Candidates\}} \sum_{n\text{-}gram \in C} Count_{clip}(n\text{-}gram)}{\sum_{C' \in \{Candidates\}} \sum_{n\text{-}gram' \in C'} Count(n\text{-}gram')}.$$

$$\mathrm{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}$$

BP is brevity penalty. this is added because the generated caption length may be smaller or greater than reference sentence. If the length is greater then bp=1 else it is calculated as exponent of 1-ratio of lengths.

$$\mathrm{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right).$$

References

- Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan: *”Show and Tell:Lessons Learned from the 2015 MSCOCO Image Captioning Challenge”*

- Marc Tanti, Albert Gatt, Kenneth P. Camilleri: *”What is the Role of Recurrent Neural Networks (RNNs) in an Image Caption Generator?”*

- Piyush Sharma, Nan Ding, Sebastian Goodman, Radu Soricut:*”Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset for Automatic Image Captioning ”*

- Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S. Zemel, Yoshua Bengio:*”Show, Attend and Tell: Neural Image Caption Generation with Visual Attention”*

Thank You