A Project Report on

Image Captioning Using Deep Neural Networks

Submitted in partial fulfillment of the requirements for the award
of the degree of

Bachelor of Engineering

in

Computer Engineering

by

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

Under the Guidance of

Prof. Sachin Malave

Department of Computer Engineering
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2019-2020

Approval Sheet

This Project Report entitled ***"Image Captioning Using Deep Neural Networks"*** Submitted by ***"Mrunal S Jadhav"(16102030), "Aditya G Joshi"(16102017),***is approved for the partial fulfillment of the requirenment for the award of the degree of ***Bachelor of Engineering*** in ***Computer Engineering*** from ***University of Mumbai***.

Prof. Sachin Malave
Head Department of Computer Engineering

Place:A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***"Image Captioning Using Deep Neural Networks"*** submitted by ***"Mrunal S Jadhav" (16102030),"Aditya G Joshi" (16102017)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Computer Engineering***,to the University of Mumbai,is a bonafide work carried out during academic year 2019-2020.

Prof. Sachin Malave
Guide

Prof. Sachin Malave
Head Department of Computer Engineering

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

____

(Signature)

____

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

Date:

Contents

List of Figures

List of Tables

Abstract

The visual world is populated with a vast number of objects, the most appropriate labeling 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 quest of connecting computer vision and natural language processing is a long way of touching the holy grail in artificial intelligence.

Image captioning is the task of generating natural language utterances based on the understanding of visual snippets of a scene.Image captioning is a much more involved task than image recognition or classification, because of the additional challenge of learning representations of the interdependence between the objects/concepts in the image and the creation of a succinct sentential narration.

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).Aided by the advances in training neural networks and large dataset available, there has been a surge in research interest attacking the image caption generation problem. 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.

Chapter 1

Introduction

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. This task is significantly harder, than the well-studied image classification or object recognition tasks, which have been a main focus in the computer vision community. Indeed, 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.

Moreover, the above semantic knowledge has to be expressed in a natural language like English, which means that a language model is needed in addition to visual understanding. 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. Therefore, image captioning is a more complicated but meaningful task in the age of artificial intelligence. The challenge of image captioning is to design a model that can fully use image information to generate more human-like rich image descriptions. 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.

1.1 Objective

To Generate an Encoder-Decoder Network Architecture with

- CNNs to produce a rich representation of the input image by embedding it into a fixed-length vector, such that this representation can be used for a variety of vision tasks.
- RNN network that obtains historical information through continuous circulation of the hidden layer, which has better training capabilities and can perform better than mining deeper linguistic knowledge such as semantics and syntax information implicit in the word sequence.

Chapter 2

Literature Review

The first significant work in solving image captioning tasks was done by Ali Farhadi[5] 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.

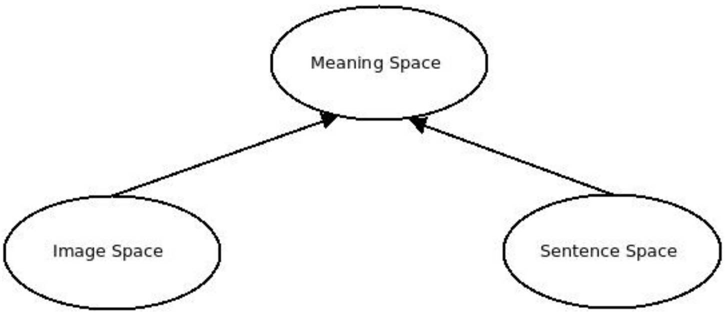


Figure 2.1: Every Picture Tells a Story: Generating Sentences from Images

With the help of mapping, similarity between the images and the sentence is evaluated, the meanings are stored as triplets of (image, action, object) and a score is evaluated by predicting the image and sentence triplets. If an image and sentence have high level of similarity in terms of the predicted triplets then they will be highly compatible and have a high score. Thus, appropriate sentences can be generated. This model has many drawbacks such as requirement of the middle meaning space and the results obtained from it are not at all highly accurate.

Various other works were introduced but more recent work use the methodology of neural networks for solving the task. With the advent of Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), a good performance was achieved and found applications in various fields of study. 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's 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.

Inorder to generate sentences it uses beam search algorithm. Beam search algorithm considers multiple alternatives and keeps them in memory. The number of alternatives kept in the memory depends upon the paramter B, beam width. The paper tries two different values of B 1 and 20.Later for competition, different values of B were tried and based on the CIDER metric they found the optimum value of B.The paper stresses on Bleu Score as Evaluation metrics. Given a machine generated translation, the Bleu score measures how good the translation is. It calculates Bleu score till 4 gram.Dropout and Ensembling gave an improvement in Bleu points and reduced overfitting along with other methods employed such as using pretrained ImageNet model and word embeddings.

On the contrary, a novel approach is followed by [6] by generating captions using visual attention.In this approach, attention is given to the most important part of the image and producing a sentence around it. In real world scenarios there is noise or clutter present in the images so unlike the traditional methods, not all the features are fed into the (RNN) but only the important and salient features are fed into the (RNN).

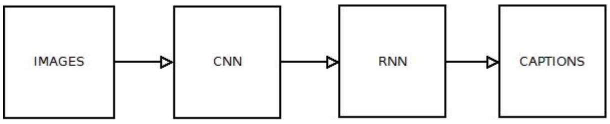


Figure 2.2: Without Attention Mechanism

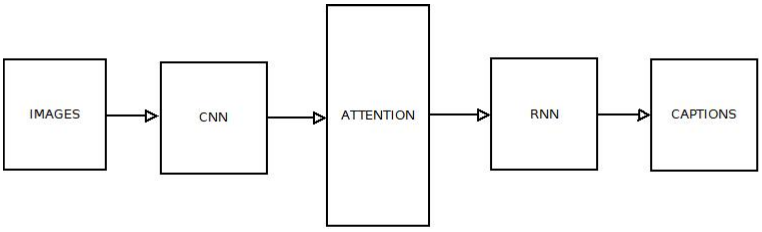


Figure 2.3: With Attention Mechanism

A better performance framework was achieved by [7] by taking visual attention as the basis of the proposal.

Chapter 3

Problem Statement

For the image captioning task, humans can easily understand the image content and express it in the form of natural language sentences according to specific needs; however, for computers, it requires the integrated use of image processing, computer vision, natural language processing and other major areas of research results. The challenge of image captioning is to design a model that can fully use image information to generate more human-like rich image descriptions.

Chapter 4

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. We propose to follow this elegant receipe by replacing the encoder RNN by a deep CNN which can produce a rich representation of input by embedding it in a fixed-length vector, such that this representation can be used for variety of tasks. Thus we develop an CNN model by pretraining it for an image classification task and using the last hidden layer as an input to the RNN decoder which uses the fixed dimensional vector representation to “decode” it to the desired output sentence. We intend to develop a single end to end network to develop more accurate feature extraction and efficiently generate textual description which can provide detailed information about the given image.

Chapter 5

Technology Stack

5.1 Platform

- Colaboratory is a Google research project created to help disseminate machine learning education and research. It's a Jupyter notebook environment that requires no setup to use and runs entirely in the cloud.

5.2 Libraries

- PyTorch is an open source machine learning library based on the Torch library, used for applications such as computer vision and natural language processing. It is primarily developed by Facebook's artificial intelligence research group
- NumPy is the fundamental package for scientific computing with Python. It contains among other things
  - a powerful N-dimensional array object
  - sophisticated (broadcasting) functions
  - tools for integrating C/C++ and Fortran code
  - useful linear algebra, Fourier transform, and random number capabilities
- Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.
- Keras 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. Keras High-Level API handles the way we make models, defining layers, or set up multiple input-output models.
- Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.
- OpenCV library for preprocessing images.

Chapter 6

Benefits for the environment

- Generate captions for images which can promote safety and protection of environment
- Determine various pollutants present in a given image and caption them so as to reduce its generation and manage it.
- Image search tools can help in finding environmental problems and its solutions.

Chapter 7

Benefits for the Society

- Helps visually impaired to understand the image by converting the captioned text into speech.
- Helps colour blind and other vision problem patients to understand image more effectively.
- Providing more accurate captions for images which can be easily misunderstood.
- Finding hidden features from keywords generated by captions.

Chapter 8

Application

The web is filled with billions of images, helping to entertain and inform the world on a countless variety of subjects. The existing NLP applications that benefit which extract insights/summary from given text data or an essay etc can be extended to people who would benefit from automated insights from images. While automatic image captioning can help solve this problem, accurate image captioning is a challenging task that requires advancing the state of the art of both computer vision and natural language processing. Elaborate applications in image captioning are :

- Help Visually Impaired:
  Much of the visual information is not accessible to those with visual impairments. A rich automated description of the image would benefit them.
- E -commerce assistant:
  Image captions, manually added by website authors using Alt-text HTML, is one way to make this content more accessible, so that a natural-language description for images that can be presented using text-to-speech systems. However, existing human-curated Alt-text HTML fields are added for only a very small fraction of web images.
  - Mapping images to natural language and vice versa would also help in medical image understanding that conveys the clinical physician that the algorithm has found something fishy in the patient by mapping the physiological parameters and images which may require further investigations.
  - Platforms like Facebook can infer directly from the image, where you are (beach, cafe etc), what you wear (colour) and more importantly what you're doing also (in a way).
  - These applications can be extended to explaining what is happening in a video frame by frame.

Chapter 9

Proposed System

The proposed system consists of an Encoder Convolutional Neural Network and a Decoder Recurrent Neural Network. The Encoder Convolutional Neural Network is used for feature extraction for a given input image. The output of the encoder network if forwarded as input to the Decoder Recurrent Neural Network. The decoder network then will use Long Short Term Memory blocks (LSTM) to predict the words from the given features and generate the captioning sentence.

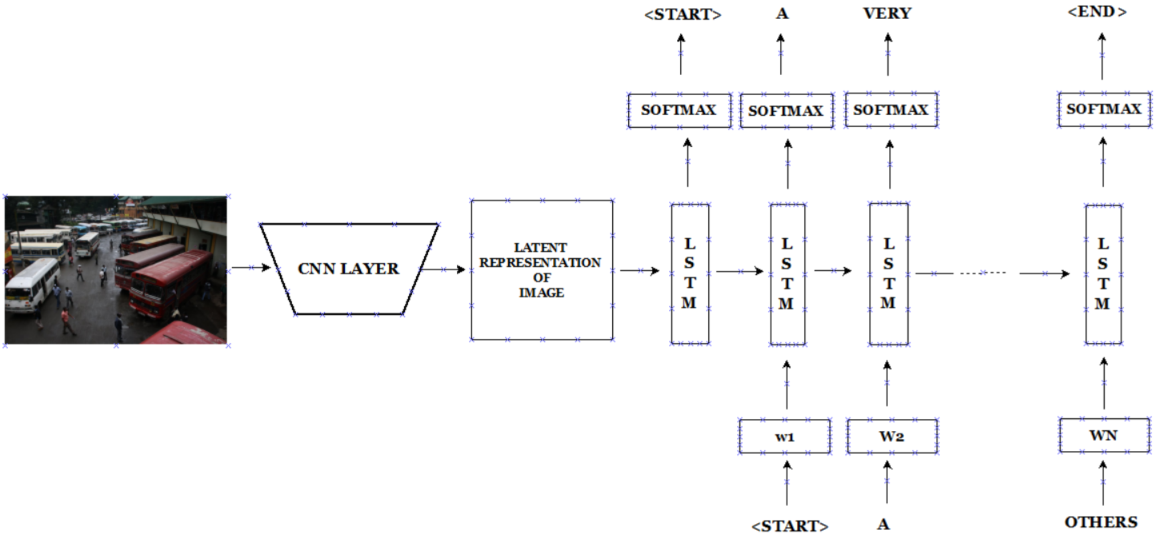


Figure 9.1: Flow Diagram

Encoder Convolutional 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 or objects in the image and be able to differentiate one from other. A deep encoder CNN will produce a rich representation of input by embedding it in a fixed-length vectors. In our proposed system we are going to use a pretrained model of resnet50 present in keras. The resnet50 model

is pretrained on Imagenet dataset and is widely used in transfer learning. Decoder Recurrent Neural Network
A recurrent Neural Network (RNN) can be thought as multiple copies of same network, each passing a message to its successor. A decoder RNN uses the last hidden of CNN layer as an input to RNN decoder which uses the fixed dimensional vector representation to decode its desired output caption. The decoder RNN uses LSTM blocks and softmax activation. The output of current recurrent unit is also forwarded to next LSTM block along with the input embedding. The start symbol is passed to the first block of RNN.

The input image is passed on to the CNN model for feature extraction. In our project CNN model is a pretrained Resnet50 model. The architecture of Resnet50 consists of two major blocks, Identity block and Convolution block. The block which will be used is decided by input and output dimensions. The architecture diagram of Resnet50 is as follows

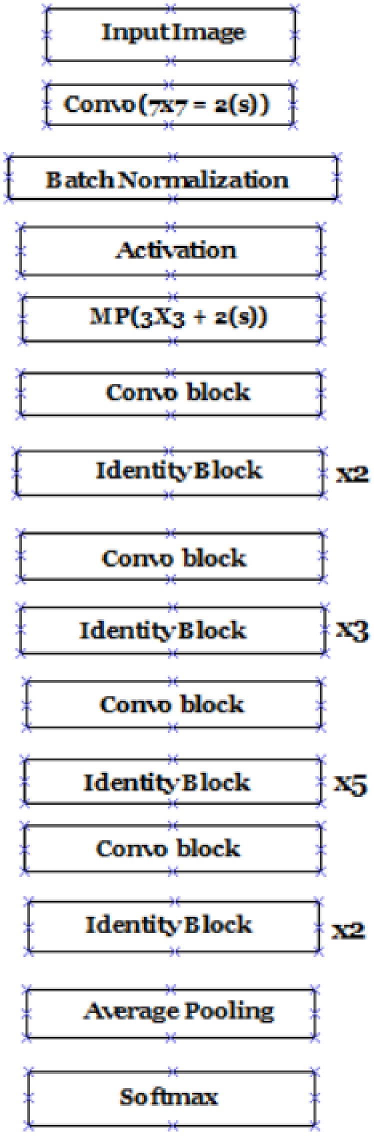


Figure 9.2: Flow of Resnet50

The output of last hidden layer of Resnet50 is passed as input to RNN model. The first recurrent block of RNN model takes this input as well as a < START > symbol input. In the successive recurrent units of RNN the output of previous block as well as the current is passed till the last recurrent unit. The softmax activation is applied on output of LSTM block to get the desired words.

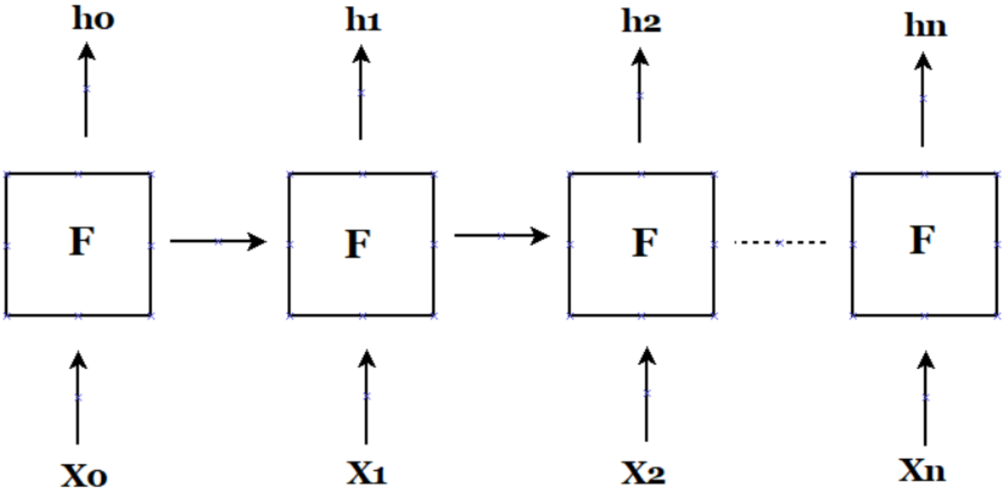


Figure 9.3: RNN expanded in time

Chapter 10

Dataset

10.1 Motivation

A large amount of work has been done on image caption generation task. Most of the significant work in solving computer vision tasks involve following prominent datasets.

- MSCOCO Dataset
  MSCOCO is a large-scale object detection, segmentation, and captioning dataset. The dataset provides 5 captions per image for image captioning. It includes 123287 color images (10^6 magnitude).
- Flickr30k Dataset
  Flickr30k is a standard dataset for achieving benchmark results for sentence based Image description.

Dataset	**No.Of Images**	**Objects per Image**	**Object Categories**	**Object per Category**	**Sentences Per Image**
MSCOCO	3,28,000	8.7	44,518	6.2	5
Flickr30k	31,783	7.7	91	27,473	5

Table 10.1: Statistics for Datasets MSCOCO and Flickr30k

- Pinterest Image Dataset : Also related to this work is the Pinterest image and sentence-description dataset (Mao et al., 2016). It is a large dataset (order of 108 examples), but its text descriptions do not strictly reflect the visual content of the associated image, and therefore can not be used directly for training image-captioning models.

An explosion of proposed solutions based on deep learning architectures have been provided on these datasets have been provided. To represent wider variety of styles, a new dataset for Image Captioning task was introduced by Google on September 5, 2018.

10.2 Google's Conceptual Caption Dataset

- Conceptual Captions, contains an order of magnitude more images than the MS-COCO dataset and represents a wider variety of both images and image caption styles.

- In contrast with the curated style of the COCO images, Conceptual Captions images and their raw descriptions are harvested from the web, and therefore represent a wider variety of styles.

- The raw descriptions are harvested from the Alt-text HTML attribute associated with web images.

- An automatic pipeline extracts, filters, and transforms candidate ¡image, caption pairs¿, with the goal of achieving a balance of cleanliness, informativeness, fluency, and learnability of the resulting captions. This pipeline is known as Flume pipeline which processes billions of web pages parallely.

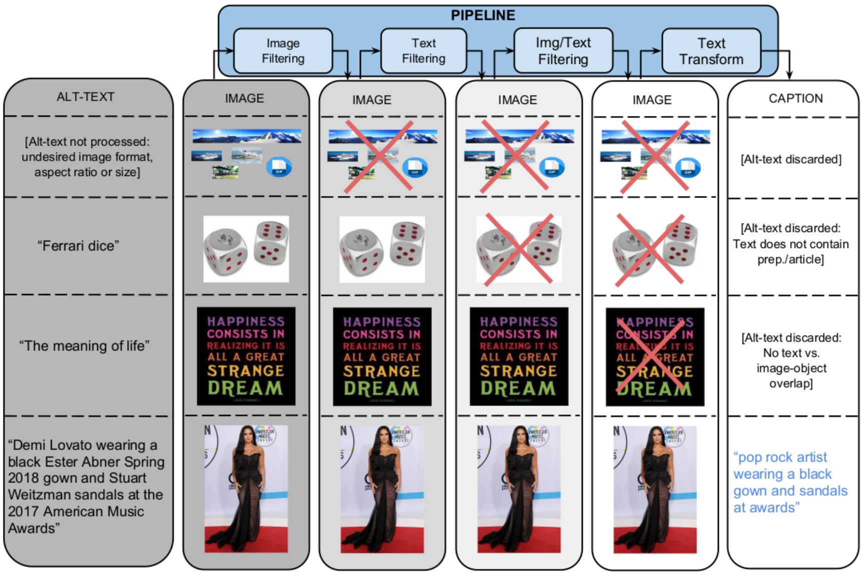

PIPELINE
Image Filtering
Text Filtering
Img/Text Filtering
Text Transform
ALT-TEXT
IMAGE
IMAGE
IMAGE
IMAGE
CAPTION
[Alt-text not processed: undesired image format, aspect ratio or size]
[Alt-text discarded]
"Ferrari dice"
[Alt-text discarded: Text does not contain prep./article]
"The meaning of life"
[Alt-text discarded: No text vs. image-object overlap]
"Demi Lovato wearing a black Ester Abner Spring 2018 gown and Stuart Weitzman sandals at the 2017 American Music Awards"
"pop rock artist wearing a black gown and sandals at awards"
HAPPINESS CONSISTS IN REALIZING IT IS ALL A GREAT STRANGE DREAM

Figure 10.1: The Flume Pipeline for Dataset Creation

1. Image Based Filtering :
   The first filtering stage, image-based filtering, discards images based on encoding format, size, aspect ratio, and offensive content.These filters discard more than 65% of the candidates.

2. Text Based Filtering :
   The second filtering stage, text-based filtering, harvests Alt-text from HTML webpages. Using Google Cloud Natural Language APIs, candidate captions have been analyzed on the heuristic have been performed such as part-of-speech (POS), sentiment/polarity, and pornography/profanity annotations. This filtering allows only 3% of the candidates to pass to the later stages.

3. Image-Text Based Filtering :
   Using Google Cloud Vision APIs labels are assigned to the images using image classifier and thus the text tokens can be mapped to the content of the image. This discards 60% of the candidates.

4. Text Transformation with Hypernymization :
   Proper names of people location and venue are difficult to learn as a part of the image captioning task. Google Cloud Natural Language APIs gives named-entity and syntactic-dependency annotations. Then the Google Knowledge Graph (KG) Search API matches the named-entities to KG entries and exploit the associated hypernym terms.

Dataset Description :

Split	Examples	Unique Tokens	Mean	StdDev	Median
Train	3,318,333	51,201	10.3	4.5	9.0
Valid	28,355	13,063	10.3	4.6	9.0
Test	22,530	11,731	10.1	4.5	9.0

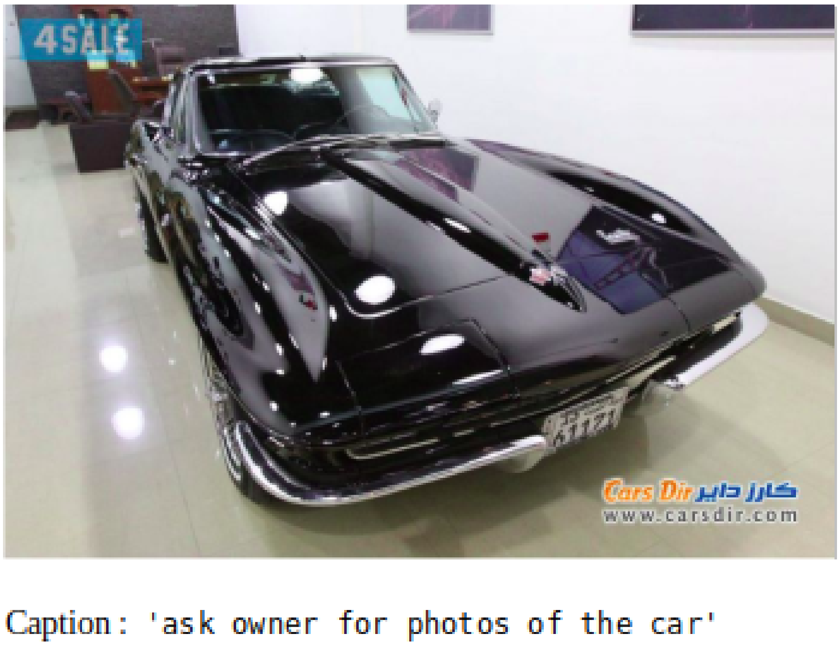
Table 10.2: Statistics of Google's Conceptual Captioning Dataset

The Hidden Test Sample hasn't been officially released and dedicated to supporting submissions at a hosted competition.

Dataset Format :
The released data is provided as TSV (tab-separated values) text files with the following columns:

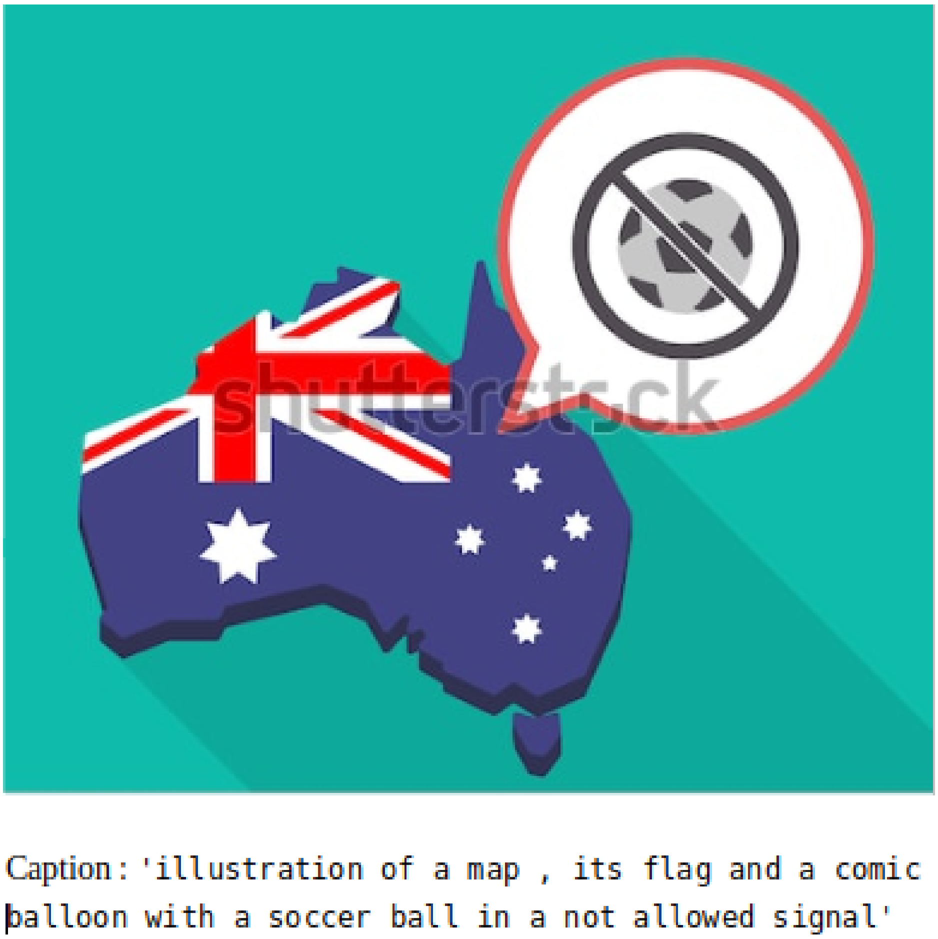
Column	Description
1	Caption. The text has been tokenised and lowercased
2	Image URL

Table 10.3: Dataset Format



Caption: 'ask owner for photos of the car'

Figure 10.2: Dataset Sample 1



Caption: 'illustration of a map , its flag and a comic balloon with a soccer ball in a not allowed signal'

Figure 10.3: Dataset Sample 2

Chapter 11

Encoding the Image

The encoding part of the network that compresses the input into a latent-space representation. It can be represented by an encoding function h=f(x). To learn this method, we propose to study and evaluate the following two methods to learn the representation and find the best fit for the dataset.

11.1 Transfer Learning on ResNet50:

Neural Style Transfer (NST) uses a previously trained convolutional network, and builds on top of that. The idea of using a network trained on a different task and applying it to a new task is called transfer learning.The availability of powerful computer vision models such as ResNet it is possible to convert the pixels of the image into high-level features with no manual feature-engineering.
The implementation of ResNet50 model trained on more than million images from the ImageNet database was a major breakthrough as it allowed training deep neural networks. As more layers are added to the model, because of the vanishing gradient problem, model weights of the first layers can not be updated correctly through the backpropagation of the error gradient. The objective of ResNet was to preserve gradient.
In ResNets, a ”shortcut” or a ”skip connection” allows the gradient to be directly back propagated to earlier layers.

Two main types of blocks are used in a ResNet, depending mainly on whether the input/output dimensions are same or different.

The Identity Block

The identity block is the standard block used in ResNets, and corresponds to the case where the input activation (say a[l]) has the same dimension as the output activation (say a[l+2]).

The Convolution Block

This type of block is used when the input and output dimensions don’t match up. The difference with the identity block is that there is a CONV2D layer in the shortcut path.

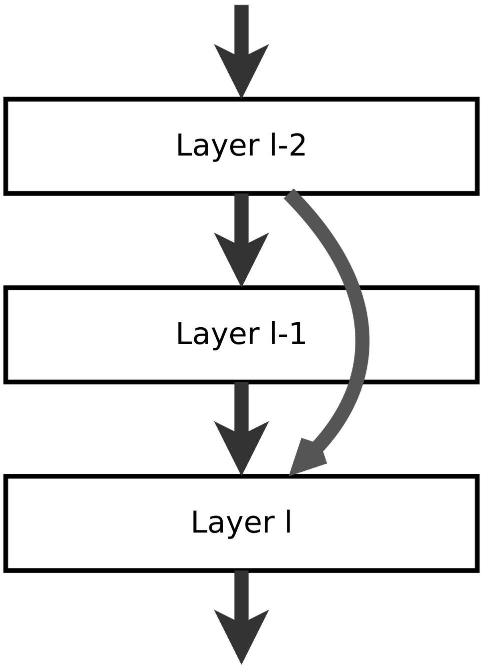


Figure 11.1: Skip Connection in ResNet

However these modules have been widely used and studied for image tasks, and are currently state-of-the art for object recognition and detection. These models extract features from the images. However, by using pre-trained model for image captioning task,we aim to evaluate the extent to which it takes the orientational and the spatial relationship of the features into consideration.

11.2 Using Autoencoders:

An autoencoder neural network is an unsupervised learning algorithm that applies backpropagation, setting the target values to be equal to the inputs. I.e., it uses y(i)=x(i). In other words, it is trying to learn an approximation to the identity function, so as to output x that is similar to x.
An encoder architecture is as follows

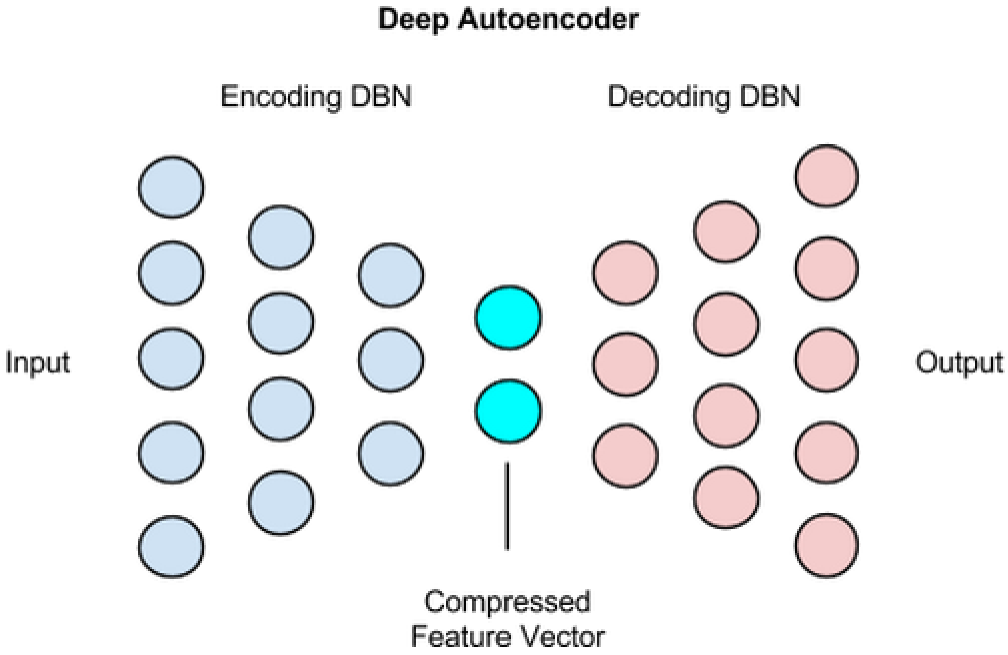


Figure 11.2: Autoencoder Architecture

As show in the figure, the autoencoder would compress the input into a lower-dimensional vector and then reconstruct the output from this representation. The vector is a compact summary of the input, also called the latent-space representation and will be passed as an input activation to our RNN network.
Along with dimensionality reduction, the intuition behind using autoencoders is that it is data-specific. Autoencoders are only able to meaningfully compress data similar to what they have been trained on. Hence, they learn features specific for our dataset.

Chapter 12

Decoder RNN

The target variable is the captions that our model is learning to predict. The output of the previous module (encoder) is fed as an input to the decoder RNN which would generate the sentences. where θ are the parameters of our model, I is an encoded image

$$\theta^{\star} = \arg\max_{\theta} \sum_{(I,S)} \log\ p(S|I;\theta),$$

representation, and S its correct transcription. Since S represents any sentence, its length is unbounded. Thus, it is common to apply the chain rule to model the joint probability over S0 ; . . . ; SN , where N is the length of this particular example as

$$\log p\ (S|I) = \sum_{t=0}^{N} \log\ p(S_t|I, S_0, \ldots, S_{t-1}),$$

Traditional Neural Networks do not preserve the information learned in previous times. RNN make use of the sequential information. RNN faces following issues

1. Long Term Dependency
2. Vanishing Gradient and Exploding Gradient

To model this probability we use Long Short Term Memory (LSTM) which is a special type of RNN. The heart of LSTM is it's cell which provides memory. The cell is made up of three types of gates

1. **Input Gate** : Input Gate which tells us that what new information we're going to store in the cell state
2. **Output Gate** : Output gate is used to provide the activation to the final output of the lstm block at timestamp 't'.
3. **Forget Gate** : Forget gate which tells the information to throw away from the cell state.

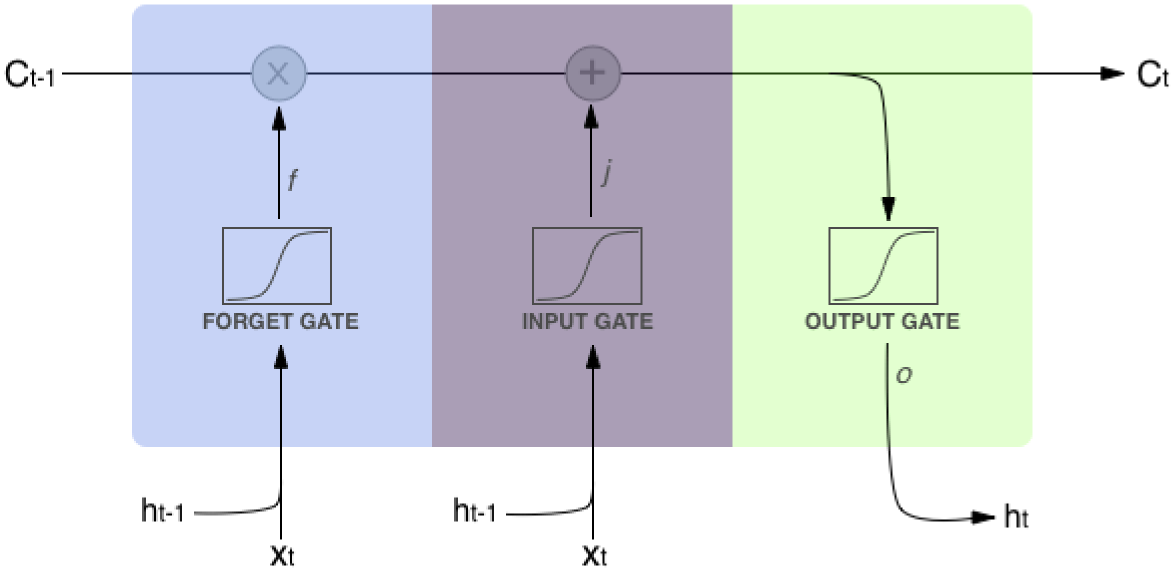


Figure 12.1: LSTM Memory Cell

The equations of the LSTM gates are :
$i_t = \sigma(w_i[h_{t-1},x_t]+b_t)$
$f_t = \sigma(w_f[h_{t-1},x_t]+b_f)$
$o_t = \sigma(w_o[h_{t-1},x_t]+b_o)$

where,

$i_t \Rightarrow$ represents input gate
$f_t \Rightarrow$ represents forget gate
$o_t \Rightarrow$ represents output gate
$\sigma \Rightarrow$ represents sigmoid function
$w_x \Rightarrow$ weight for the respective gate(x) neurons
$h_{t-1} \Rightarrow$ weight for previous lstm block(at timestamp t-1)
$x_t \Rightarrow$ input at current timestamp
$b_x \Rightarrow$ Biases for the respective gates(x)

The equations for the cell state, candidate cell state and the final output are :

$\tilde{ct}$ = tanh(w_c[h_{t-1},x_t]+b_c)
c_t=f_t * c_{t-1} + i_t * $\tilde{ct}$
h_t=o_t * tanh(c^t)

c_t ⇒ cell state(memory) at timestamp(t)
$\tilde{ct}$ ⇒ represents candidate for cell state at timestamp(t)

*Note : * represents the element wise multiplication of the vectors.*

Lastly, we filter the cell state and then it is passed through the activation function which predicts what portion should appear as the output of current LSTM unit at timestamp t.

We can pass this ht the output from current lstm block through the softmax layer to get the predicted output(yt) from the current block. The overall architecture is as follows :

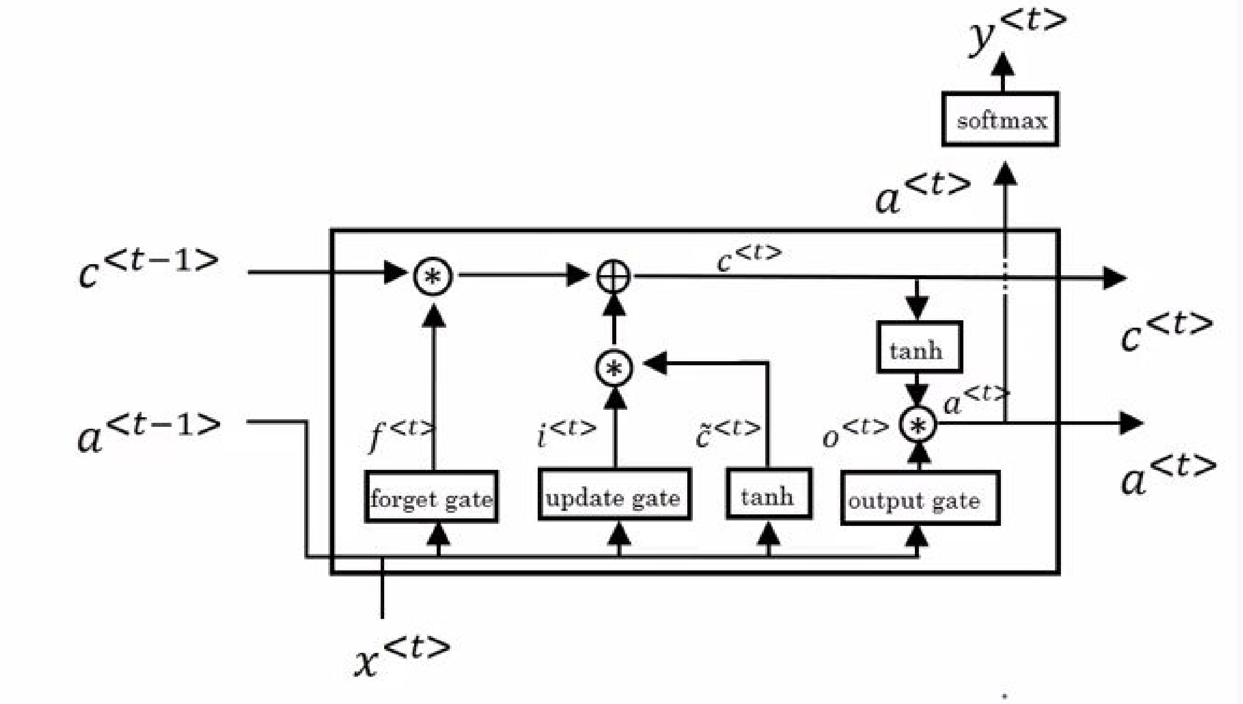


Figure 12.2: Detailed Gates of LSTM

Bibliography

[1] Show and Tell: Lessons Learned from the 2015 MSCOCO Image Captioning Challenge — Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan

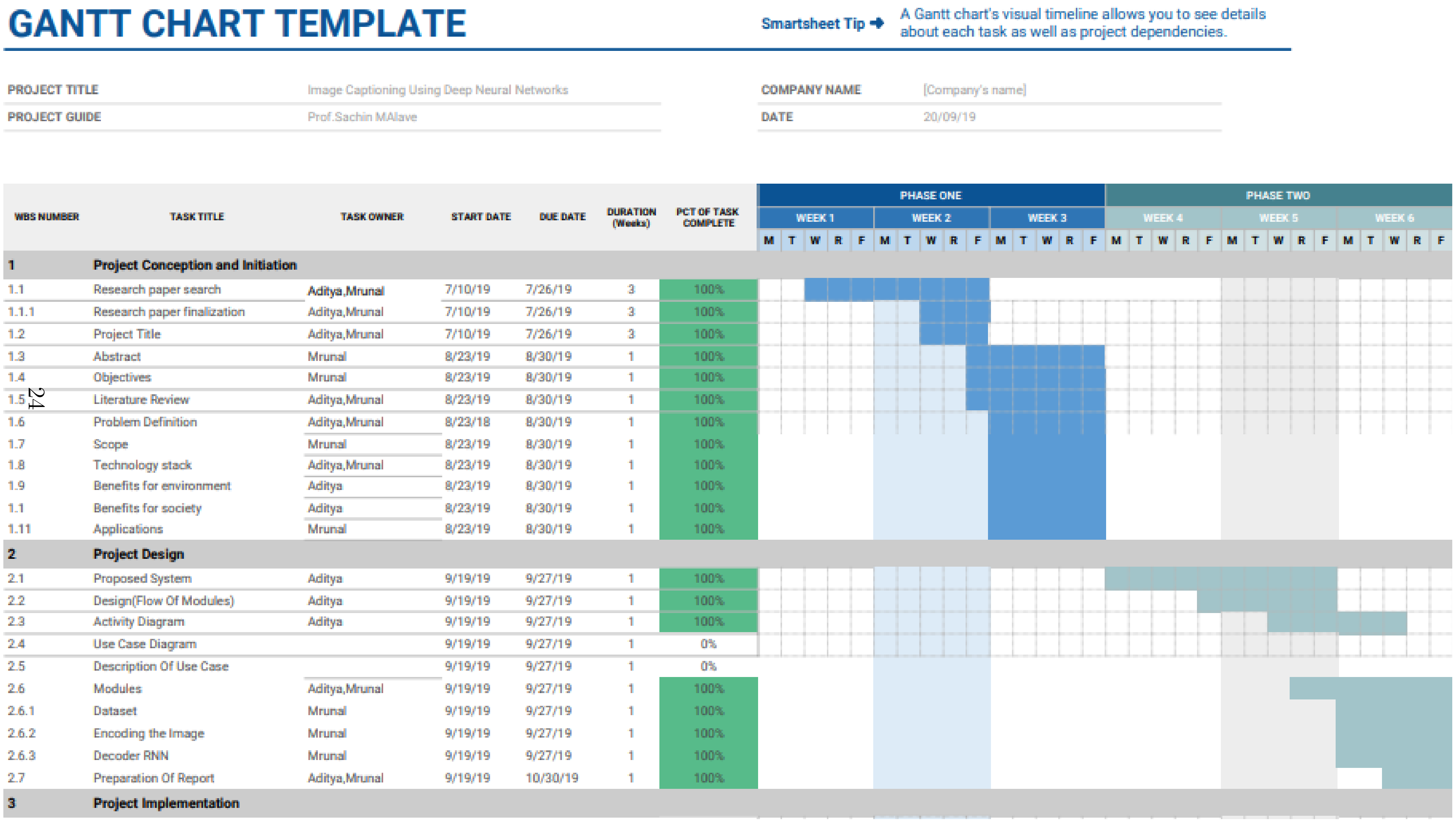
[2] Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning — Piyush Sharma, Nan Ding, Sebastian Goodman, Radu Soricut.

[3] Image Captioning Based on Deep Neural Networks — Shuang Liu, Liang Bai, Yanli Hu and Haoran Wang

[4] Transforming Auto-encoders— G. E. Hinton, A. Krizhevsky S. D. Wang

[5] Farhadi A. et al. (2010) Every Picture Tells a Story: Generating Sentences from Images. In: Daniilidis K., Maragos P.,Paragios N. (eds) Computer Vision – ECCV 2010. ECCV 2010. Lecture Notes in Computer Science, vol 6314. Springer, Berlin, Heidelberg

[6] Show, Attend and Tell: Neural Image Caption Generation with Visual Attention — Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, Yoshua Bengio ; Proceedings of the 32nd International Conference on Machine Learning, PMLR 37:2048-2057, 2015



GANTT CHART TEMPLATE

Smartsheet Tip ➜ A Gantt chart's visual timeline allows you to see details about each task as well as project dependencies.

PROJECT TITLE	Image Captioning Using Deep Neural Networks
PROJECT GUIDE	Prof.Sachin MAlave
COMPANY NAME	[Company's name]
DATE	20/09/19

WBS NUMBER	TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION (Weeks)	PCT OF TASK COMPLETE
1	**Project Conception and Initiation**					
1.1	Research paper search	Aditya,Mrunal	7/10/19	7/26/19	3	100%
1.1.1	Research paper finalization	Aditya,Mrunal	7/10/19	7/26/19	3	100%
1.2	Project Title	Aditya,Mrunal	7/10/19	7/26/19	3	100%
1.3	Abstract	Mrunal	8/23/19	8/30/19	1	100%
1.4	Objectives	Mrunal	8/23/19	8/30/19	1	100%
1.5	Literature Review	Aditya,Mrunal	8/23/19	8/30/19	1	100%
1.6	Problem Definition	Aditya,Mrunal	8/23/18	8/30/19	1	100%
1.7	Scope	Mrunal	8/23/19	8/30/19	1	100%
1.8	Technology stack	Aditya,Mrunal	8/23/19	8/30/19	1	100%
1.9	Benefits for environment	Aditya	8/23/19	8/30/19	1	100%
1.1	Benefits for society	Aditya	8/23/19	8/30/19	1	100%
1.11	Applications	Mrunal	8/23/19	8/30/19	1	100%
2	**Project Design**					
2.1	Proposed System	Aditya	9/19/19	9/27/19	1	100%
2.2	Design(Flow Of Modules)	Aditya	9/19/19	9/27/19	1	100%
2.3	Activity Diagram	Aditya	9/19/19	9/27/19	1	100%
2.4	Use Case Diagram		9/19/19	9/27/19	1	0%
2.5	Description Of Use Case		9/19/19	9/27/19	1	0%
2.6	Modules	Aditya,Mrunal	9/19/19	9/27/19	1	100%
2.6.1	Dataset	Mrunal	9/19/19	9/27/19	1	100%
2.6.2	Encoding the Image	Mrunal	9/19/19	9/27/19	1	100%
2.6.3	Decoder RNN	Mrunal	9/19/19	9/27/19	1	100%
2.7	Preparation Of Report	Aditya,Mrunal	9/19/19	10/30/19	1	100%
3	**Project Implementation**					

Figure 12.3: Phase 1 : Gantt Chart

Acknowledgement

We have great pleasure in presenting the report on **Image Captioning Using Deep Neural Networks** We take this opportunity to express our sincere thanks towards our guide **Prof. Sachin Malave** Department of Computer Engineering, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Prof. Sachin Malave** Head of Department,Computer Engineering, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof.Amol Kalugade** BE project co-ordinator, Department of Computer Engineering, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Student Name1:Mrunal S Jadhav
Student ID1: 16102030

Student Name2: Aditya G Joshi
Student ID2: 16102017