

# Image Annotation Using Neural Networks

VII<sup>th</sup> SEM PROJECT  
IIIT - ALLAHABAD

MENTOR  
DR. SONALI AGARWAL

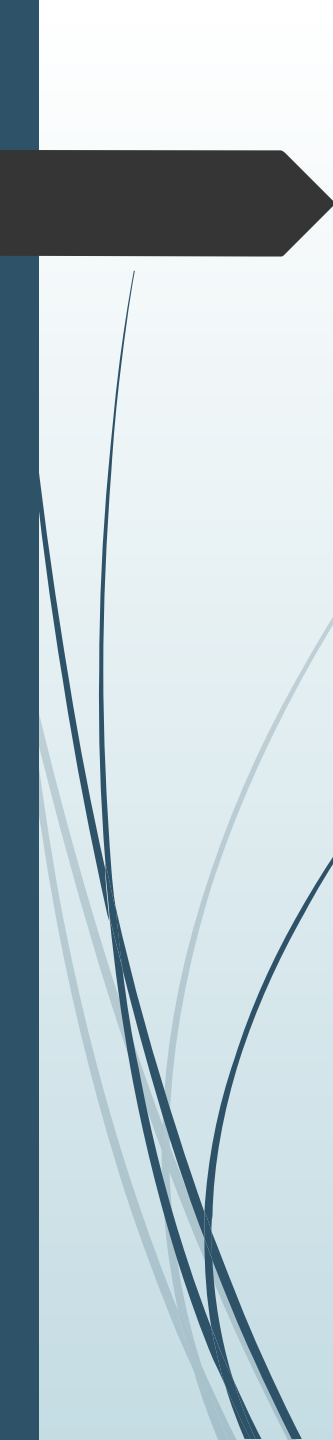




# Introduction

Humans just by having a quick look at an image can acquire immense amount of details from the image regarding the objects present in it, the background and the context of image. However , the same task is a lot difficult for computers and visual models to gain insights from an image.

We plan to present a model that provide a natural language description of given input image.

- 
- In this project we aim to work towards generating sentence description for any image independent of any hard coding of template and visual concepts.
  - We plan to devise a model such that it should be free of assumptions about specific hard-coded templates, rules or categories and instead rely on learning from the training data.



Description - *"Two dogs are playing in snow"*



# Motivation

- Importance of this project lies when a user has a slow internet connection and all the images on the web page can't be loaded, the summary (description) of all images to give him an idea what the image is about.
- Also an extension of this project can benefit visually impaired people as we can convert this summary of image to speech so that visually impaired people could hear it.

# Literature Survey

We investigated 15 different papers and we formulate our literature review in subparts given below:

## **Dense image annotations:**

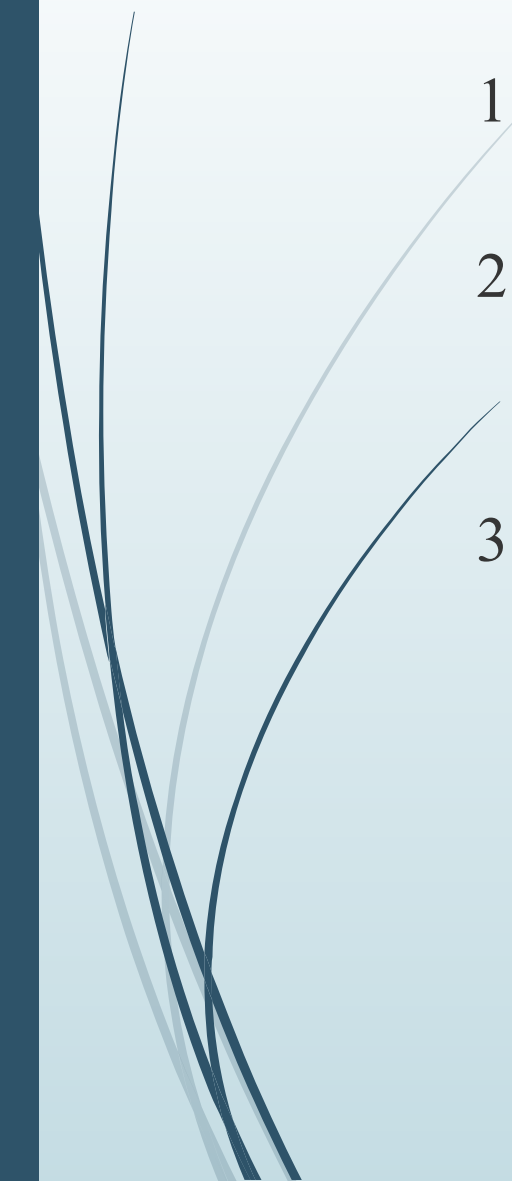
1. The focus of previous works were on correctly labeling scenes, objects and regions with a fixed set of categories.
2. Our focus is on richer and higher-level descriptions of regions.

## **Generating descriptions:**

1. Previous approaches pose the task as a retrieval problem, where where training annotations are broken up and stitched together.
2. Other approaches generate image captions based on fixed templates that are filled based on the content of the image or generative grammars.
3. We didn't chose this approach because it limits the variety of possible outputs.



## Neural networks in visual and language domains:

- 
1. Multiple approaches have been developed for representing images and words in higher-level representations.
  2. On the image side, Convolutional Neural Networks (CNNs) have recently emerged as a powerful class of models for image classification and object detection.
  3. On the sentence side, our work takes advantage of Recurrent Neural Network.




# Problem Definition

- The main aim of the project is to generate annotation for images . The major challenge is designing a model that generate these annotations that appropriately describe the contents of the image and its representation in the domain of natural language.
- The model should learn from the training images and their annotations and should not rely on hard coded rules and categories.





# Proposed Approach



We propose to make use of training dataset of images and their corresponding description to learn about relations between language and visual data.

1. We will use a deep convolutional neural network model with the aim to convert an image into corresponding vector that would be fed into RNN.
2. We will then develop a Recurrent Neural Network architecture that makes use of the data vector for an input image and generates its natural language sentence description.
3. Finally, we will train the model on this alignment model and evaluate its performance on a new dataset of images using appropriate metrics like BLEU score.

Input Image



**Convolutional  
Neural Network**



Image feature  
vector



**Recurrent Neural  
Network**



Natural Language  
Description



# Activity Time Chart

	Mid-Sem		End-Sem	
	Phase-I 15Aug- 31 Aug	Phase-II 01Sep- 15Sep	Phase-III 25Sep- 20Oct	Phase-IV 20Oct- 20 Nov
LITERATURE SURVEY				
PROBLEM IDENTIFICATION				
IDENTIFY AN APPROACH				
DATA COLLECTION				
USING CNN				
DEVELOPING RNN				
TRAINING AND TESTING				
EVALUATION				



# Hardware and Software Requirements

## Hardware-

- Machine with GPU sufficient enough for training of over 1,00,000 images.

## Tools , Libraries and Programming Language -

- Linux Based OS.
- Python Programming language.
- Numpy
- Scipy
- argparse
- Other relevant python libraries.



# Implementation

# Convolutional Neural Network

A simple Convolutional Neural Network is a sequence of layers that is used generally for analyzing visual imagery. The Hidden layers present in CNN are:

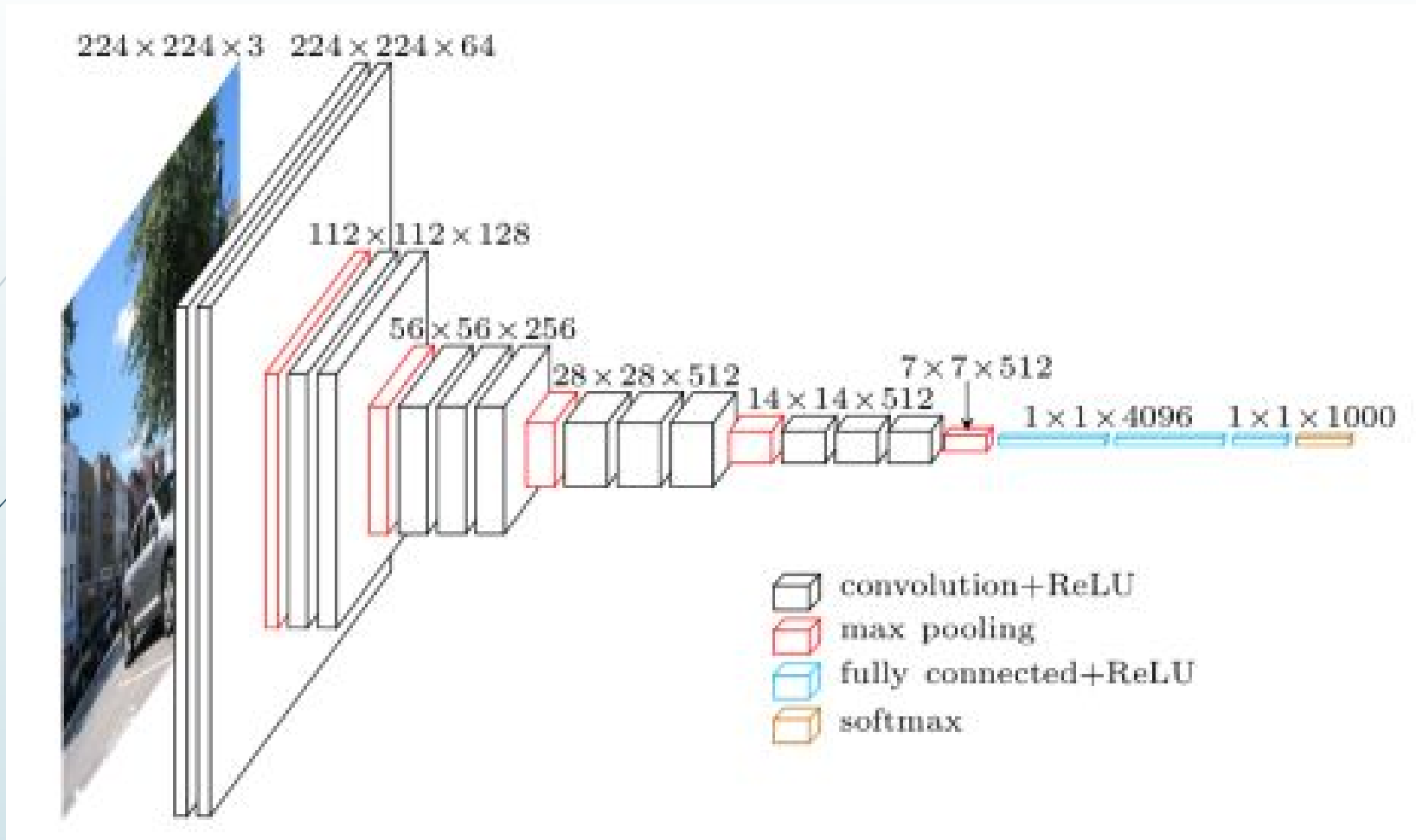
**CONVOLUTIONAL LAYER** : We convolute the image with filters and result is a stack of images filtered images.

**RELU (Rectified Linear Unit)** : To make everything negative to zero.

**POOLING LAYER** : Downsampling the image. We select a window size and then we move our window through filtered image and then take the max value of them mostly.

**FULLY CONNECTED LAYER** : Output is a vector that contains vote for different classes in our dataset.





**Architecture of VGG-16 Pretrained CNN**

# Details about VGG-16

- Runner-up in ILSVRC 2014.
- Network contains 16 CONV/FC layers and features a homogeneous architecture.
- CONV layers perform 3x3 convolutions with stride 1 and pad 1.
- POOL layers perform 2x2 max pooling with stride 2 (and no padding).
- Uses a Softmax loss function at the end.
- In the Softmax classifier, the function mapping  $f(x_i; W) = Wx_i$  stays unchanged, but it interprets these scores as the unnormalized log probabilities for each class and loss function has the following form

$$L_i = -f_{yi} + \log \sum_j e^{f_j}$$

# Recurrent Neural Network

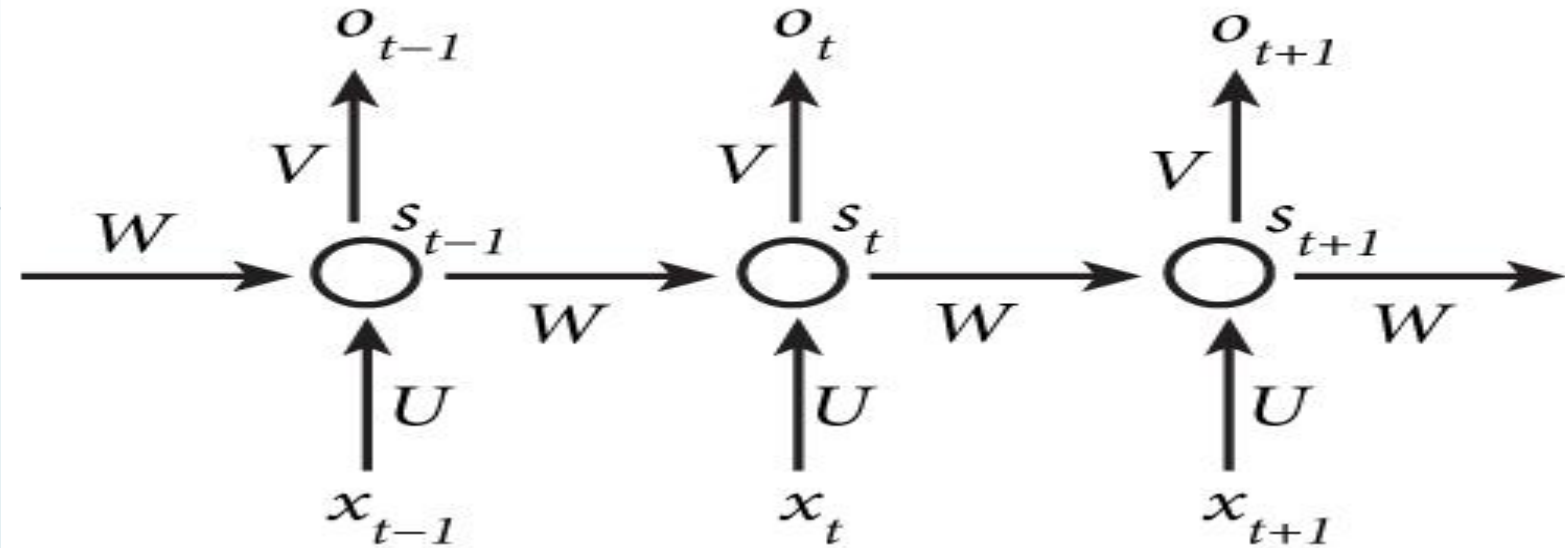
- The idea of RNN is to make use of **sequential information**.
- To predict the next word, the **knowledge of previous words** that have come before is required.
- **Recurrent** because they do the same task for every word of the sequence, and the output depends on the previous computations.
- RNN have their own **memory** which stores the information about what has been calculated so far.



Given  $m$  words, probability of observing the sentence is -

$$P ( w_1, w_2, \dots, w_m ) = P( w_i | w_1, \dots, w_{i-1} )$$

i.e, Probability of a sentence is the product of probabilities of each word given the words that came before it, eg, the probability of the sentence “**He went to buy some chocolate**” would be the probability of “**chocolate**” given “**He went to buy some**”, multiplied by the probability of “**some**” given “**He went to buy**”, and so on.



$$\mathbf{b}_v = \mathbf{W}_{hi}[\text{CNN}_{\theta_c}(\mathbf{I})]$$

$$\mathbf{h}_t = \mathbf{f}(\mathbf{W}_x \cdot \mathbf{x}_t + \mathbf{W}_h \cdot \mathbf{h}_{t-1} + \mathbf{1} + \mathbf{b}_v)$$

$$\mathbf{y}_t = \text{softmax}(\mathbf{W}_d \cdot (\mathbf{h}_t))$$

# Data Set Description

The Dataset that we will be working on are -

## 1- Flickr8k<sup>[4]</sup>

It contains 8000 images and is provided by University of Illinois. Each image has 5 captions in the dataset. The dataset can be obtained from - [http://nlp.cs.illinois.edu/HockenmaierGroup/Framing\\_Image\\_Description/KCCA.html](http://nlp.cs.illinois.edu/HockenmaierGroup/Framing_Image_Description/KCCA.html)

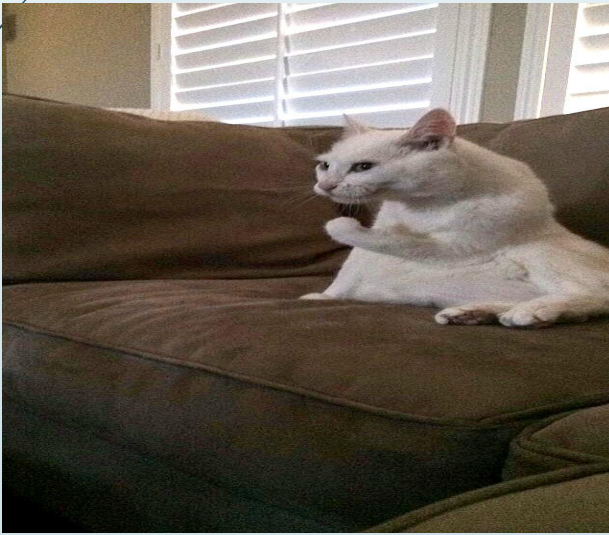
## 2- MS COCO<sup>[5]</sup>

Microsoft COCO is a large-scale object detection, segmentation, and captioning dataset. It consists of 1,23,000 images. It has fewer categories but more instances per category, which enables better learning and makes this a richer dataset. Each image has 5 caption. The dataset can be obtained from - <http://cocodataset.org/#download>

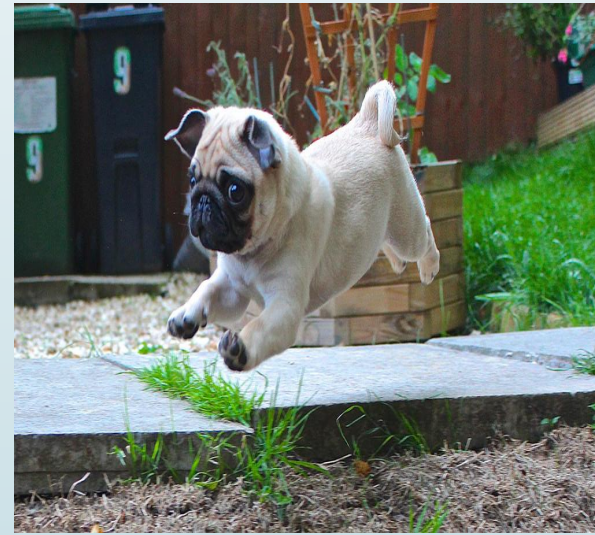
# Results and Comparison

The model was trained separately on 2 datasets - flickr8k and MSCOCO . We perform qualitative evaluation as well as we calculate the evaluation metric BLEU score. BLEU Score evaluates a candidate sentence by measuring how well it matches a set of five reference sentences written by humans which is provided in the dataset.

The results obtained on MSCOCO Dataset is shown below -

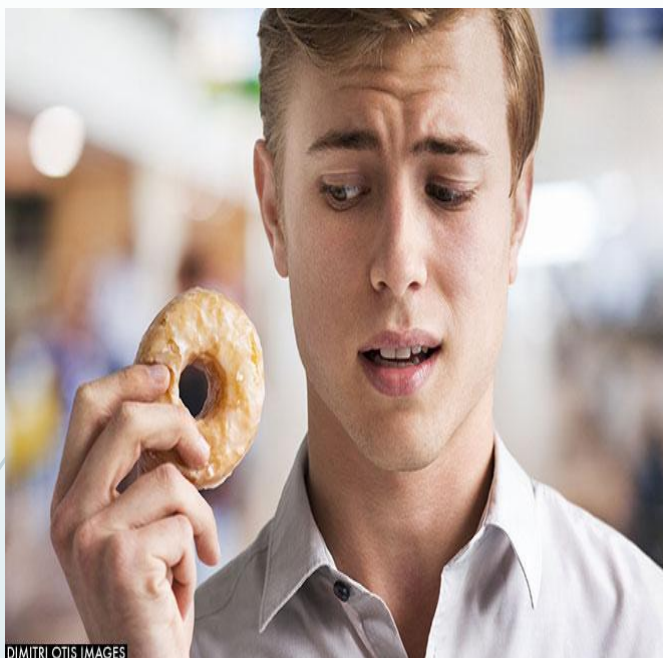


A close up of a cat laying on a bed



A dog that is standing in the grass





A close up of a person holding a doughnut.



A beach with a lot of chairs and umbrellas.

### Qualitative Evaluation -

The model generates sensible descriptions of images (see above). Although , the captions generated not always give accurate results but most of the times , the captions fairly describe the image .In general, we find that a relatively large portion of generated sentences (60% with beam size 7) can be found in the training data. This fraction decreases with lower beam size; For instance, with beam size 1 this falls to 25%, but the performance also deteriorates in terms of the captions generated and their relevance with the image.



## Comparison with other work -

We compare the results obtained by our model with other popular image captioning models . The comparison is done on the basis of BLEU Score which we obtained for different models from MSCOCO Captioning Challenge . We make a comparison on the basis of 4 Bleu Score - B-1, B-2, B-3, B-4 where each B-n is a Bleu Score that uses up to n-grams. The value of B-n varies from 0 to 100 .

MSCOCO Dataset				
Model	B-1	B-2	B-3	B-4
Nearest Neighbor	48.0	28.1	16.6	10.0
LRCN	62.8	44.2	30.4	--
Google	71.3	54.2	40.7	30.9
Our Model	64.9	46.4	32.1	23

Also , on Flickr8k dataset , we obtained Bleu Score as -

B-1 = 50.6 , B-2 = 32.3, B-3 =19.9 , B-4 = 12.6

# Conclusions

We developed a model that generates captions for images using neural networks. We used a convolutional neural network VGGNet model to convert an image into corresponding vector to be fed into RNN. We then developed a Recurrent Neural Network architecture that makes use of the data vector for an input image and generates its natural language sentence description. Finally, we trained the model and evaluated its performance on MSCOCO and Flickr8k dataset using appropriate metrics like BLEU score. We observed that our model works at par with other state of art models in the field of image captioning and yet it is more fast and simple.



# Future Scope

- Our approach consists of two separate models. Going directly from an image sentence dataset to image level annotations as part of a single model trained end-to-end remains an open problem.
- Apart from this , instead of RNN , its more complex variant LSTM could be used which is more difficult to implement and takes longer time in training but is more efficient .

# References

1. Generating Text with Recurrent Neural Networks. *I. Sutskever, J. Martens, and G. E. Hinton. (2011)*
2. Rich feature hierarchies for accurate object detection and semantic segmentation. *Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik. (2014)*
3. Connecting Modalities: Semi-supervised Segmentation and Annotation of Images Using Unaligned Text Corpora. *Richard Socher, Li Fei-Fei*
4. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. (2014)*
5. Microsoft coco: Common objects in context. arXiv preprint arXiv:1405.0312  
*T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick*



# Thank you

Team Members :-

Yash Jain

IIT2014043

Anupam Jaiswal

IIT2014038

Shivam Awasthi

IIT2014155