

# DeepVision Tool

---



An Interactive GUI & Server Solution for  
Information Extraction from Image

# Table Of Contents

---



## Introduction

Problem Statement | Scope | Solution  
Project Progress | Previous vs New  
Implementation Details | Screenshots  
Current Outcomes | Deliverables  
Assumptions & Constraints

## Requirement Analysis

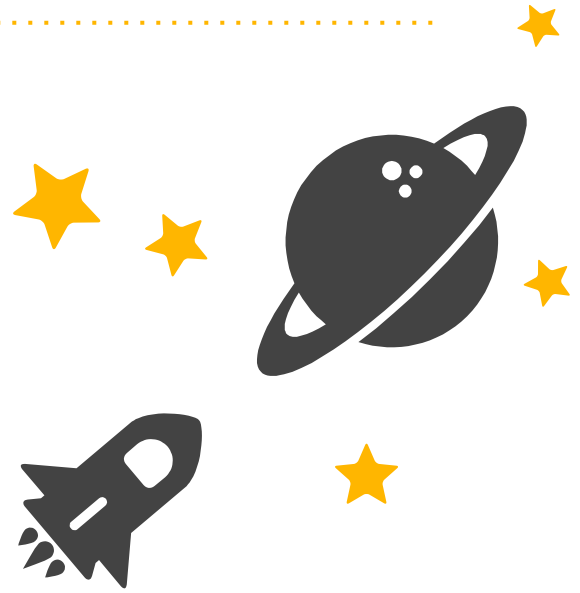
Existing Research  
Standards Used

## Methodology Adopted

Investigative Techniques

## Design Specifications

Model Architectures



# Introduction

# Problem Statements

---

1

- Preprocessing Steps Required With ML as feature engineering.
- Are GUI deployment solution for Deep Learning models effective.
- Visual QA & Activity Classification information extraction capabilities needed.
- Applications often requires chatbot as a helper tool.

## Goals

---

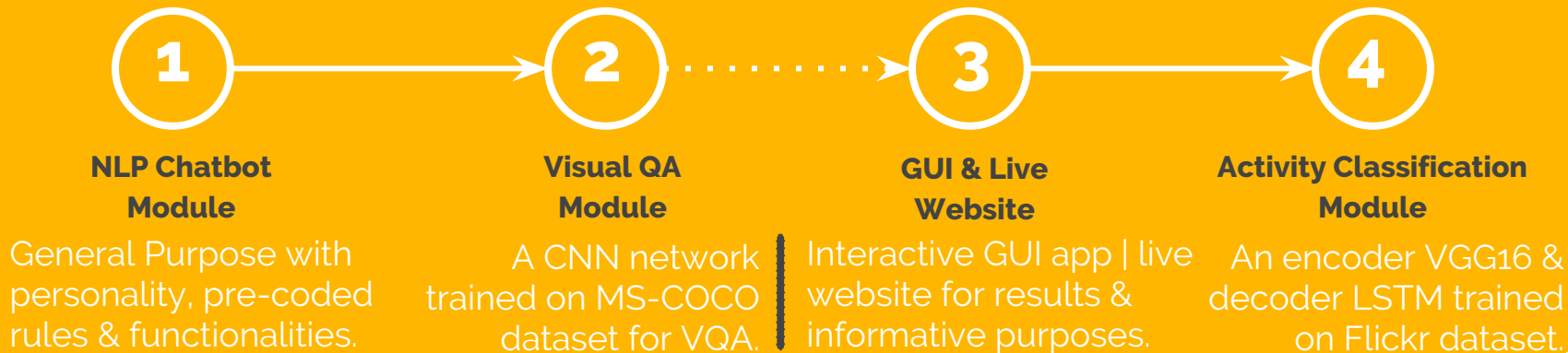
- Fully deployable solution for information extraction with scalability.
- Incorporating helper chabots and its effective practices.
- Optimal model processes for eliminating feature engineering.

# Solution



- A GUI application with chatbot capabilities for information extraction from images with Deep Learning Models.
- A server based online solution for less powerful machines.

## Project Progress



# Implementation Details

---



## Tkinter GUI

Event Driven Programming based GUI which triggers modules, help page as a subprocess.

## Website

Facilitate test image uploading, Informatory website containing all theoretical & implementation details.

---

## Visual QA Module

CNN network trained on MS-COCO dataset.

Gives probability of all possible solutions.

## NLP Chatbot Module

Seq2Seq model trained on Reddit cleaned dataset.

With personality, pre-coded information & general purpose usage.

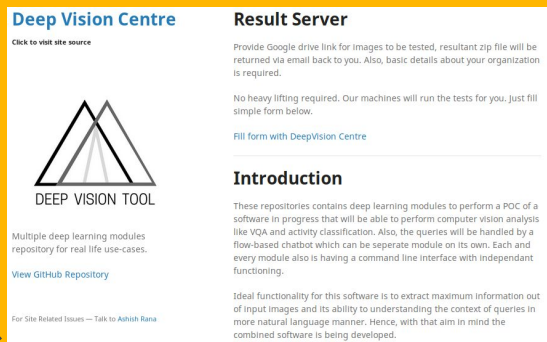
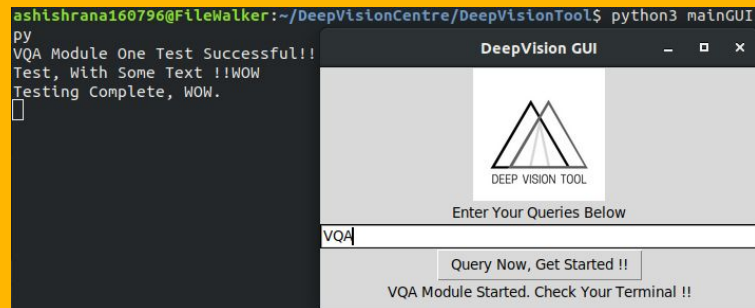
## Context Classification Module

CNN + LSTM encoder-decoder trained on Flickr 8K dataset. Generates a summarizing statement.

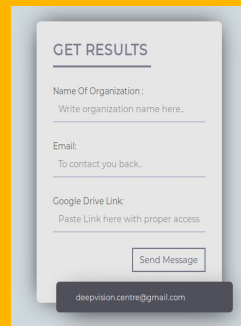
# Application ScreenShots



GUI  
Application



Website with Form  
for uploading Test  
Images.



Received Form  
Via Email.

# Current Outcomes & Deliverables

---



Two deployment techniques explored and implemented, GUI & client-server based.  
Three different deep learning models trained for three different modules.  
Independent functionality of each module is achieved.

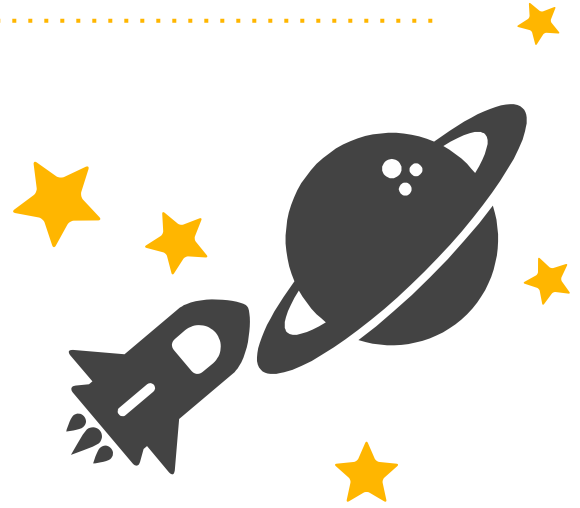
## Assumptions & Constraints

---



- Models are scalable for large testing datasets.
- DOS & DDOS attacks are not considered while developing website.
- User will only extract one kind of feature at a time i.e. one module at a time from GUI.
- Single query at a time for given module.
- User is aware about opening help page.
- Standard dataset images with no random noises.
- Version consistency for APIs & libraries used amongst modules.
- Different regularization & initialization aren't tested with limited GPU access.





# Requirement Analysis

# Existing Literature

---



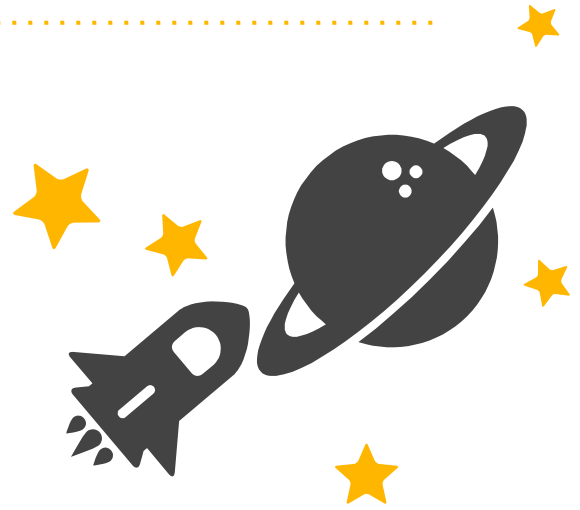
- 'Visual7W' dataset based on the MS COCO for Visual QA, largest gap in performance recorded.
- 'Visual Madlibs' dataset with fill-in-the-blanks approach focus on Natural Language Description, CNN+LSTM used but no compositional natural language understanding.
- DAQUAR dataset from NYU indoor dataset. For spatial relation finding of Visual QA.
- Neural Image-QA for caption generation, CNN + RNN trained end-to-end.
- 3-CNN architecture for encoding, composing question words & classification learning for candidate answer words.

## Standards Used

---



Project is made in compliance to IEEE standards with documentation, SRS, SQA, V&V and design specifications being the main focus..



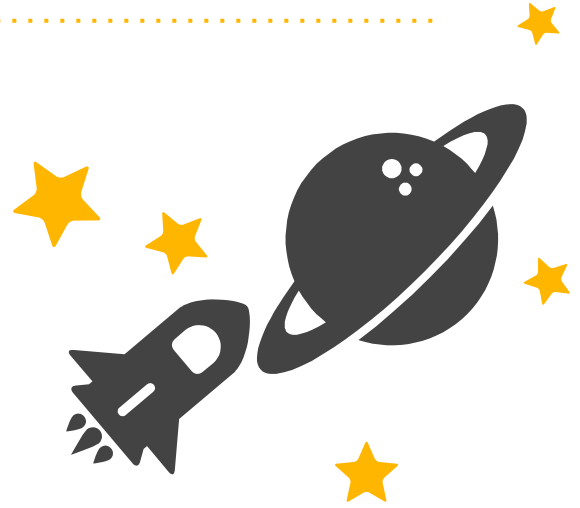
# Methodology Adopted

# Investigative Techniques

---

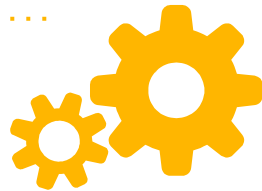


- **Caption Generator:** Apply NNs to image captioning problem with proposed MLPs with words vectorized for representation, image itself conditioned the linguistic output. CNN+LSTM model backpropagation error governs changes.
- **Visual QA Module:** First language embeddings are extracted out via RNN model to generate free form answers. Image feature extraction, question understanding, answer generation governs our further changes.
- **Chatbot Module:** Either we can use a statistical approach or a deep learning architecture approach. Instead flow based hardcoded approach is more effective as application helpers.



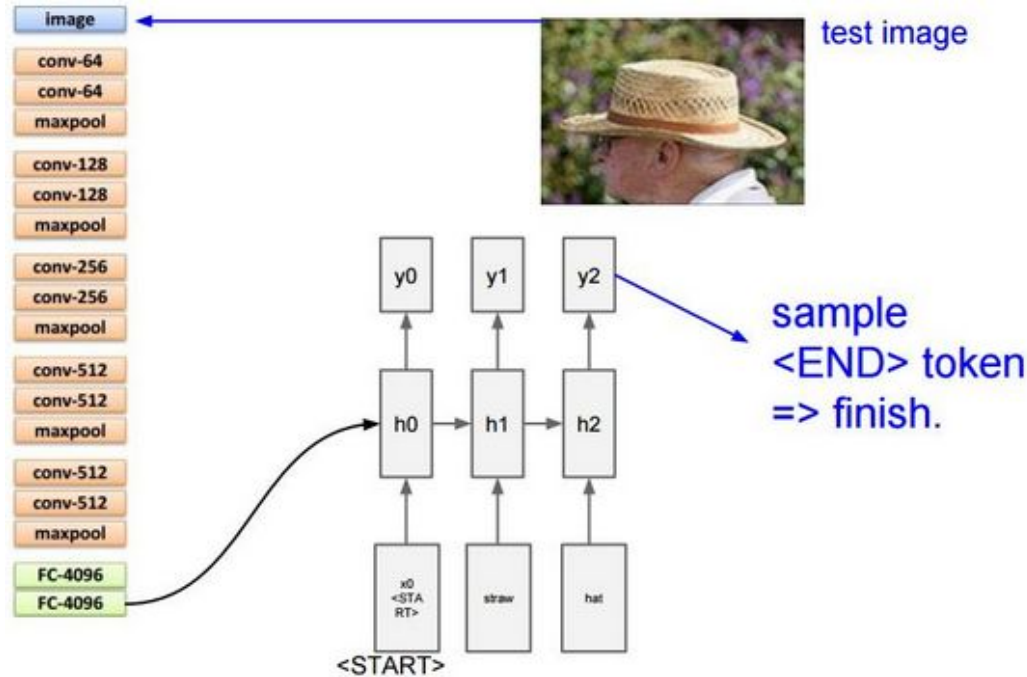
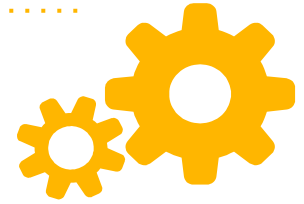
# Design Specification

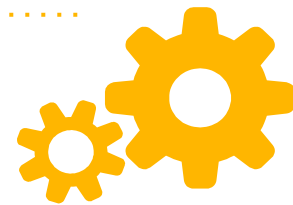
# Model Architectures



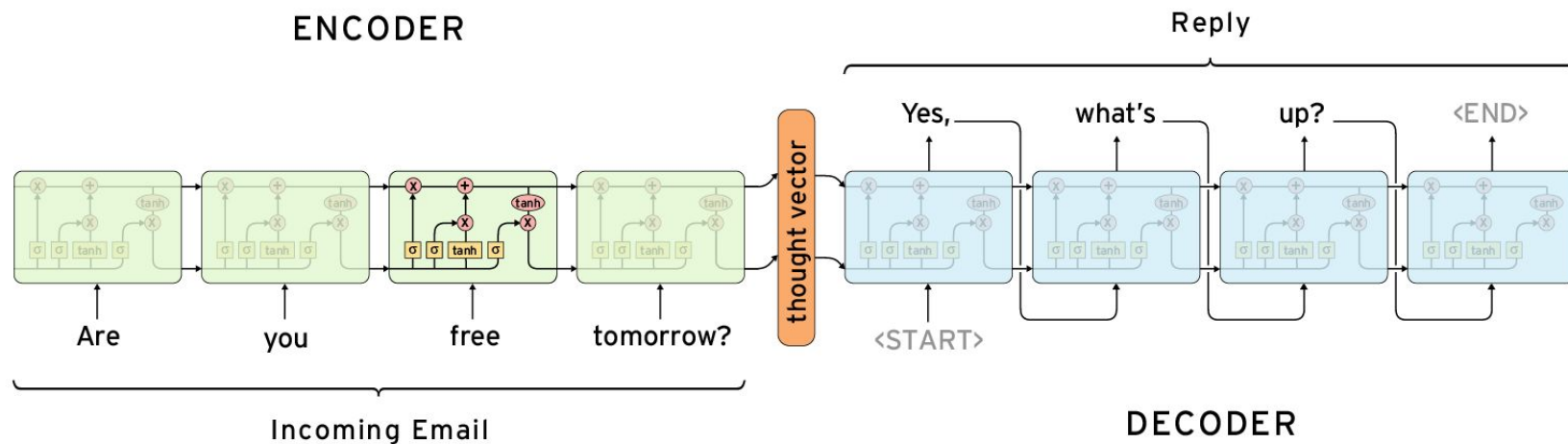
**VGG CNN Architecture with 138M parameters**

# A VGG Encoder with LSTM Decoders in a RNN.





# Seq2Seq Model Architecture with LSTM units





“

*A little learning is dangerous thing.  
Drink Deep or taste not the Pierian  
Spring.*

*Thank You !!*