



An Interative GUI & Server Solution for Information Extraction from Image





#### Introduction

Problem Statement | Scope | Solution

Project Progress | Previous vs New

Implementation Details | Screenshots

**Current Outcomes | Deliverables** 

**Assumptions & Constraints** 

#### **Requirement Analysis**

**Existing Research** 

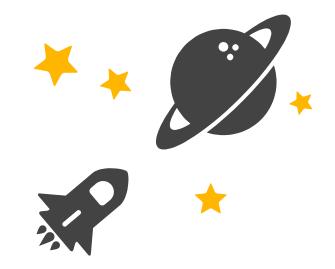
Standards Used

#### **Methodology Adopted**

**Investigative Techniques** 

#### **Design Specifications**

**Model Architectures** 



# Introduction

- 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**





Module

personality, pre-coded rules & functionalities.

Module

A CNN network trained on MS-COCO dataset for VQA.

Website

Interactive GUI app | live website for results & informative purposes.

**Activity Classification** Module

An encoder VGG16 & decoder LSTM trained on Flickr dataset.

# **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



#### **Deep Vision Centre Result Server** Click to visit site source Provide Google drive link for images to be tested, resultant zip file will be returned via email back to you. Also, basic details about your organization No heavy lifting required. Our machines will run the tests for you. Just fill simple form below. Fill form with DeepVision Centre Introduction DEEP VISION TOOL These repositories contains deep learning modules to perform a POC of a software in progress that will be able to perform computer vision analysis like VQA and activity classification. Also, the gueries will be handled by a Multiple deep learning modules flow-based chatbot which can be seperate module on its own. Each and repository for real life use-cases. every module also is having a command line interface with independant

combined software is being developed.

Ideal functionality for this software is to extract maximum information out of input images and its ability to understanding the context of queries in more natural language manner. Hence, with that aim in mind the

View GitHub Repository

Website with Form for uploading Test Images.



Received Form Via Fmail

#### **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 1

- 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**



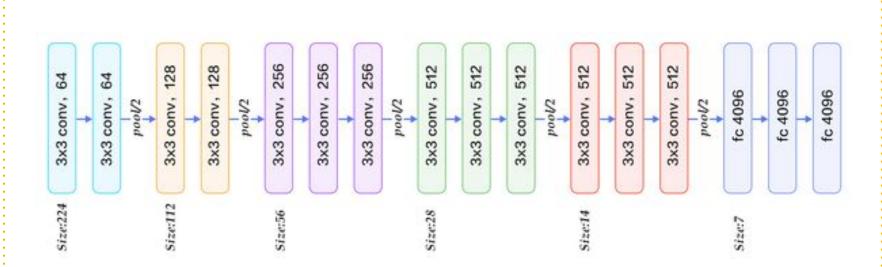
- Caption Generator: Apply NNs to image captioning problem with proposed MLPs with words vectorized for representation, image itself conditioned the linguistic output. CNN+LSTM model backpropagation error governs changes.
- Visual QA Module: First language embeddings are extracted out via RNN model to 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**

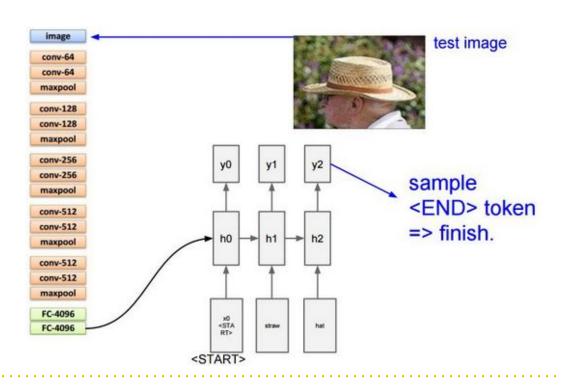




**VGG CNN Architecture with 138M parameters** 

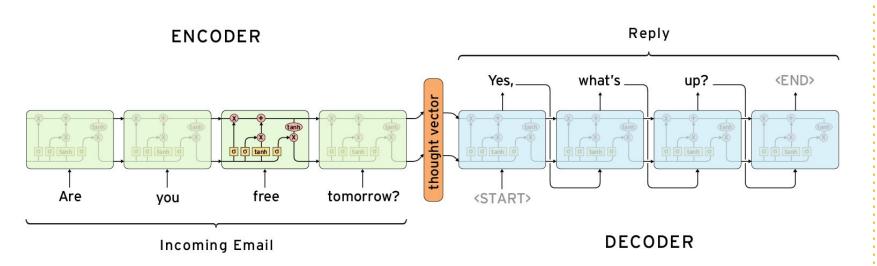
#### A VGG Encoder with LSTM Decoders in a RNN.







### **Seq2Seq Model Architecture with LSTM units**





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

Thank You!!