

Nikitha Kona-18

Prudhvi Suggula-37

Submitted by Team 11

Big Data Analytics & Apps Increment 2

Spring 2017

# Project Objectives

The objective of this project is to design an application with interactive GUI where the user can choose a picture and all the classes in the picture are identified with a 2D line around it.

## 1.1 Significance

There are ample applications that name different objects within a given picture along with the scores or probability. But there are very few applications that detect the exact position of the object in an image along with a 2D line around.

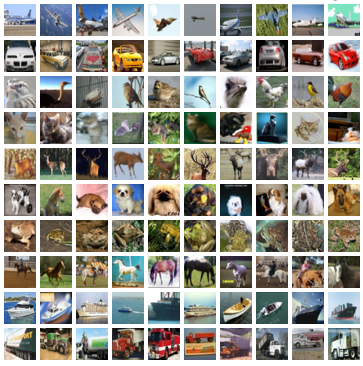
## Features

The user can choose an image from his gallery and after uploading the picture the user can see 2D line prominent around all the classes in the picture. Our application can detect if there are more than 3 classes in the same picture and all the objects can be detected accurately.

# 2.Approach

## 2.1 Data Sources

The CIFAR-10 dataset comprises of 60000 32x32 shading pictures in 10 classes, with 6000 pictures for each class. There are 50000 preparing pictures and 10000 test pictures. The dataset is partitioned into five preparing clusters and one test group, each with 10000 pictures. The test clump contains precisely 1000 haphazardly chose pictures from each class. The preparation groups contain the rest of the pictures in arbitrary request, however some preparation clusters may contain a bigger number of pictures from one class than another. Between them, the preparation clumps contain precisely 5000 pictures from each class. Here are the classes in the dataset, and 10 irregular pictures from each:



## 2.1 Analytic Tools

The analytic tools used in this project are Spark,Python,Intellij that helps to develop our application. Spark provides parallel execution a runs the program 100x faster than the traditional map reduce or 10x faster than the disk because of the in-memory execution in spark.

Due to the availability of various Deep learning libraries, graph processing libraries python is the preferred language.

## 2.2 Analytical Tasks

The primary task involved in our application is Image segmentation. Image segmentation partition the image into various pixels. The main objective of segmentation to represent the given image in a better depiction for easy analysis.

## 2.3 Expected Input and Output

The input would be an image and the output would be an image with 2D line around the respective classes. The below image shows the input and the image beside it shows the required output.



## 2.4 Algorithms

The algorithm used in our project is Convolution Neural Network(CNN). CNN are a class of neural network that are prominently used in image recognition and segmentation. They can identify the objects, faces present in the given image. Convolution is used to extract features from the input image. They preserve useful information and this is uses for further learning.

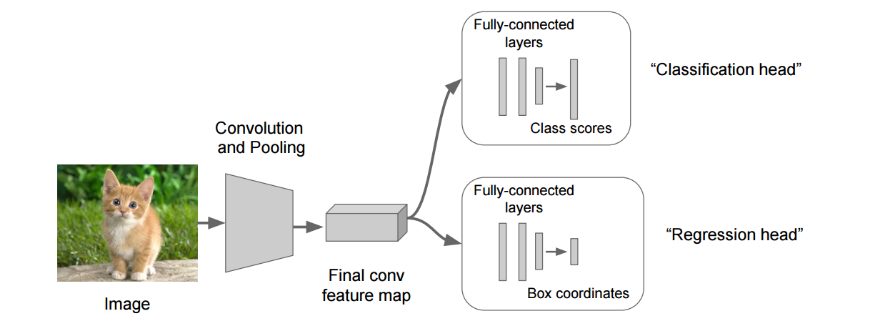
## **2.5 Inception Model**

In practice, very few people train an entire Convolutional Network from scratch (with random initialization), since it is moderately uncommon to have a dataset of adequate size. Rather, it is normal to pretrain a ConvNet on an expansive dataset (e.g. ImageNet, which contains 1.2 million pictures with 1000 classifications and then use the ConvNet either as an initialization or a fixed feature extractor for the task of interest3.Related Work.Alex net (CNN) is the landmark network in the history of computer vision and started the era of deep learning. We are using the weights of the same network , Implementing the concept of transfer learning by adding the some layers on top of it. The paper “ImageNet Classification with Deep Convolutional Neural Networks ” can be the perfect citation to this project.

ConvNet acts as fixed feature extractor. Take a ConvNet pretrained on ImageNet, expel the last completely associated layer (this current layer's yields are the 1000 class scores for an alternate undertaking like ImageNet), then treat whatever is left of the ConvNet as a settled component extractor for the new dataset. In an AlexNet, this would figure a 4096-D vector for each picture that contains the initiations of the concealed layer promptly before the classifier. We call these components CNN codes. It is essential for execution that these codes are ReLUd (i.e. thresholded at zero) in the event that they were likewise thresholded amid the preparation of the ConvNet on ImageNet (as is typically the case). When you extricate the 4096-D codes for all pictures, prepare a straight classifier (e.g. Direct SVM or Softmax classifier) for the new dataset.

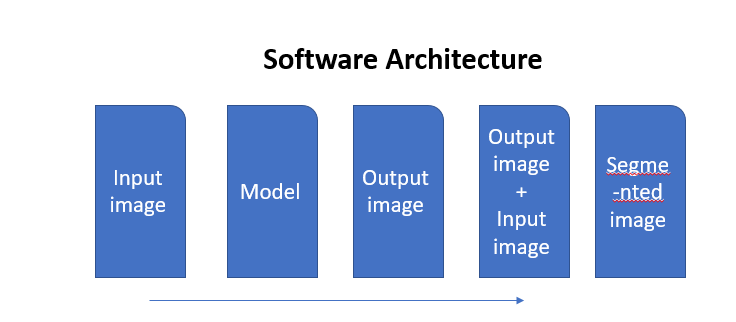
## **2.6 Bounding Boxes**

The below image depicts the architecture of our implementation. Given an image we use convolution and pooling to extract features from different scales and we perform pooling for dimensionality reduction and after achieving the final convolution feature map we have two fully connected layers one that predicts the class score which can be done by using classifiers such as softmax and the next one is regression head that produces coordinated for the bounding boxes.



# 4.Application Specifications

## 4.1 Software Architecture

  
  
  
We take (m\*n\*3) color images and they are converted to 224\*224\*3 ,Resized images are then given to the model . The model is trained to the customized class of our requirement. Capturing Activation information of last layers can help in segmentation of input image .

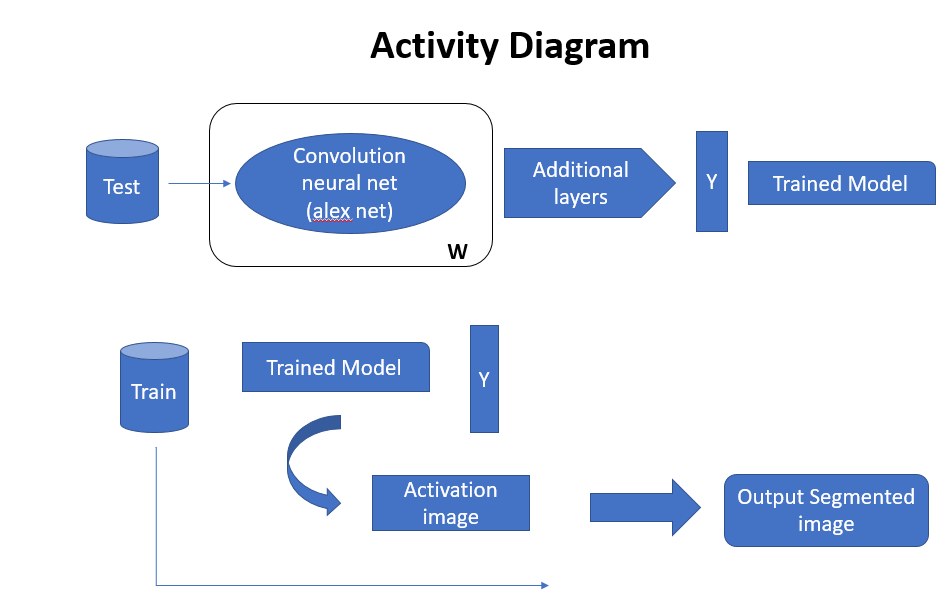
S

## 4.2 Features, Workflow & Technologies

### 4.2.1 Activity Diagram

### The below activity diagram shows how testing and training data are fed to the CNN and how the

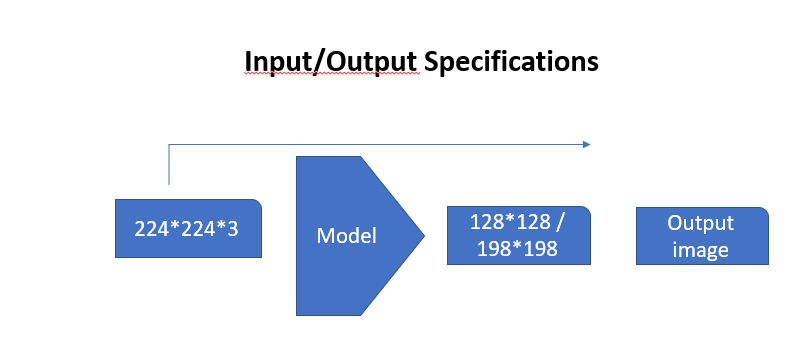
Model is trained and how the activation image is obtained.



### 4.2.2 Feature Specification

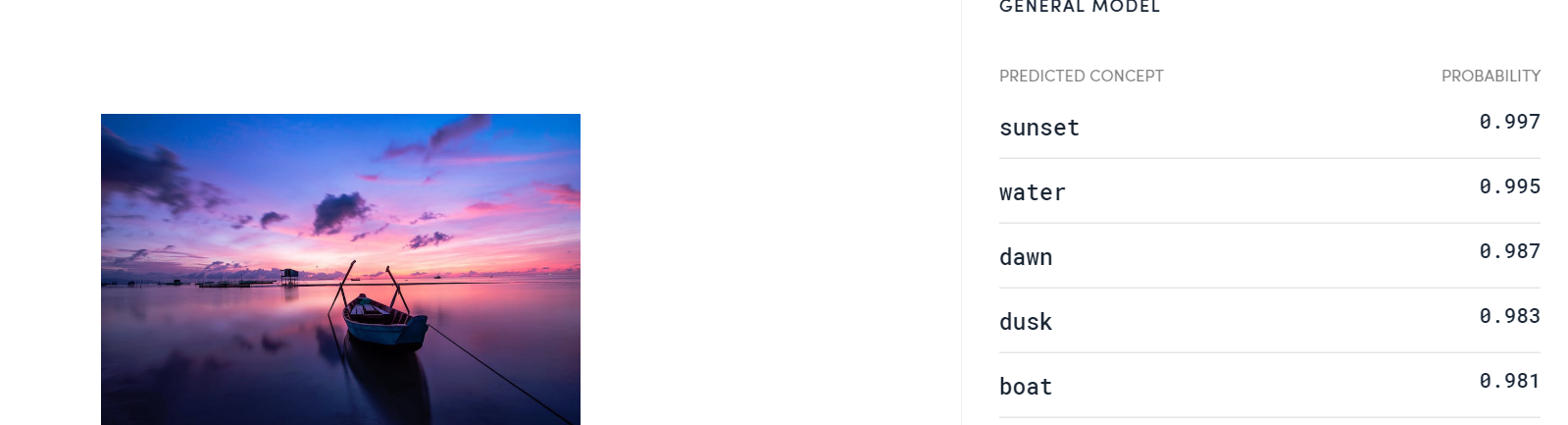
The user can choose an image from his gallery and after uploading the picture the user can see 2D line prominent around all the classes in the picture. Our application can detect if there are more than 3 classes in the same picture and all the objects can be detected accurately.

### 4.2.3 Operation Specification(Input/Output)



## 4.3Existing Applications

There are many applications that detect objects in each image with the corresponding scores and the probability. One such application is “Clarifai” that detects various objects in an image and there are a wide range of services this API provide to detect colour,apparel,faces and many more.



URL: <https://www.clarifai.com/>

# 5.Implementation

## 5.1 Implementation using clarifai API

The user can give the image and all the objects or the classes in the image are detected. We have taken an image and sent that to the clarifai API which gives all the classes in the given image. The below screenshots display the code snippet that describes the objects in the image with corresponding scores.

Code Snippet



Scores Output

The below snippet displays the relevant objects with the corresponding scores

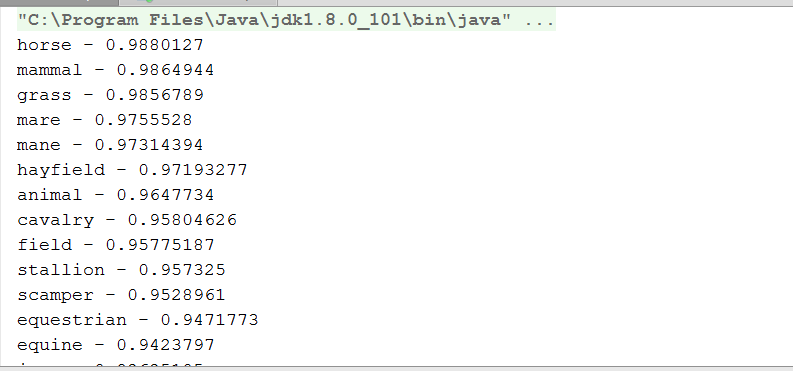


Image Output

The output image consists of annotations of the related classes in the image. The image snippet is displayed below.



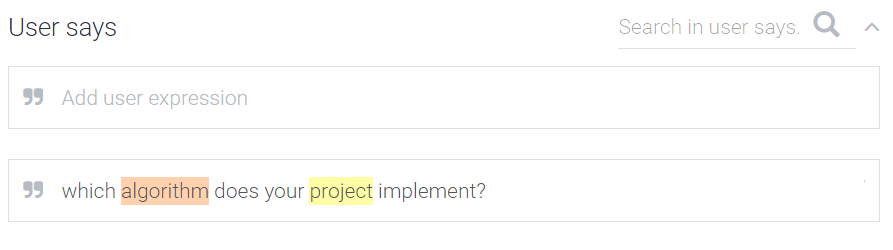
## 5.2 Google Conversation API

We have used google conversation API for interaction and we have created intents and entities and an agent to interact with the user.

## 5.2.1 Intents Creation

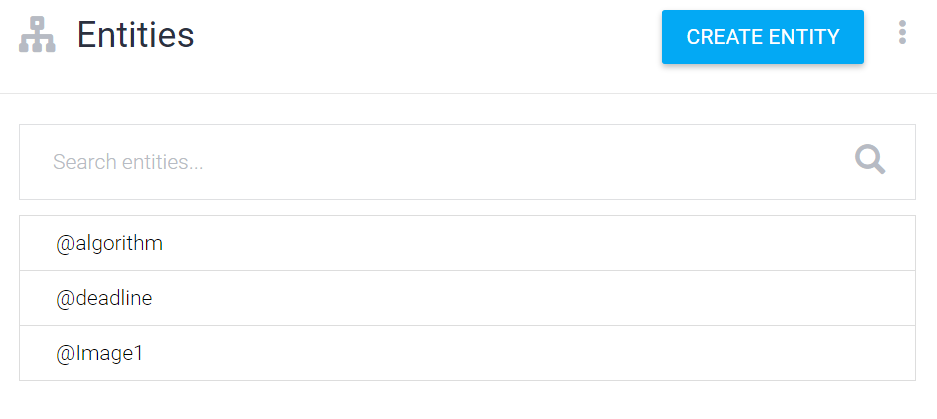
An **intent** represents a mapping between what a user says and what action should be taken by your software.

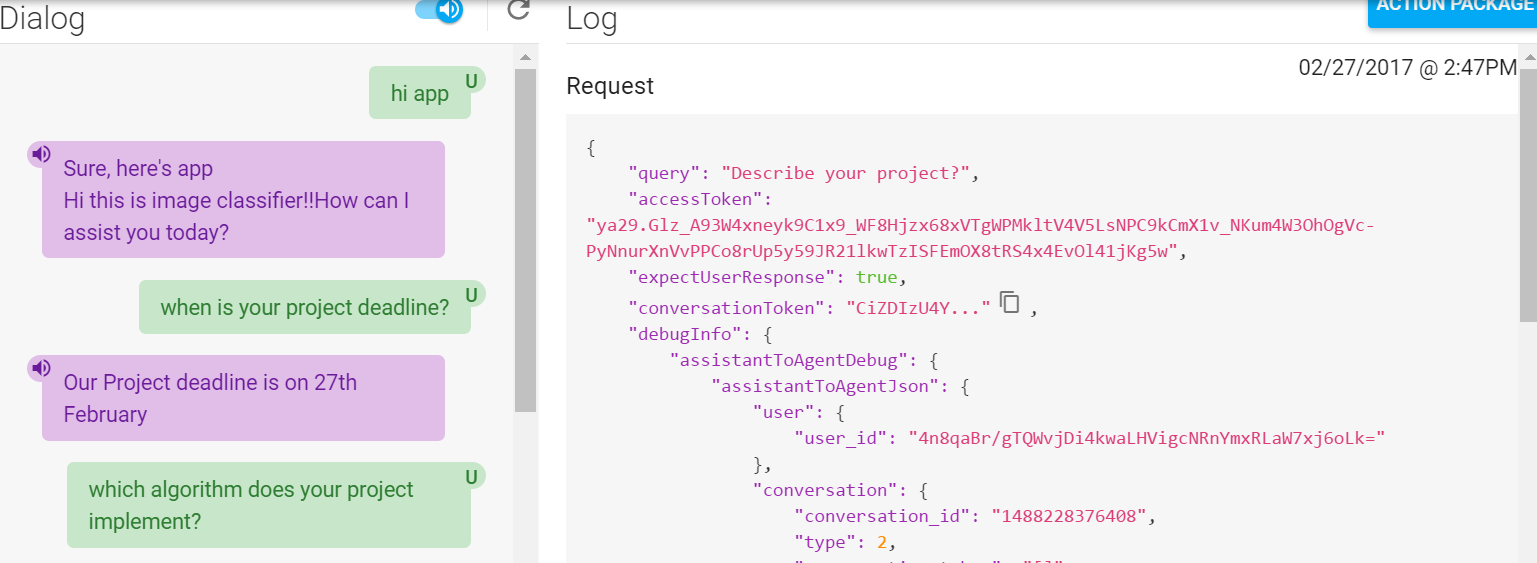


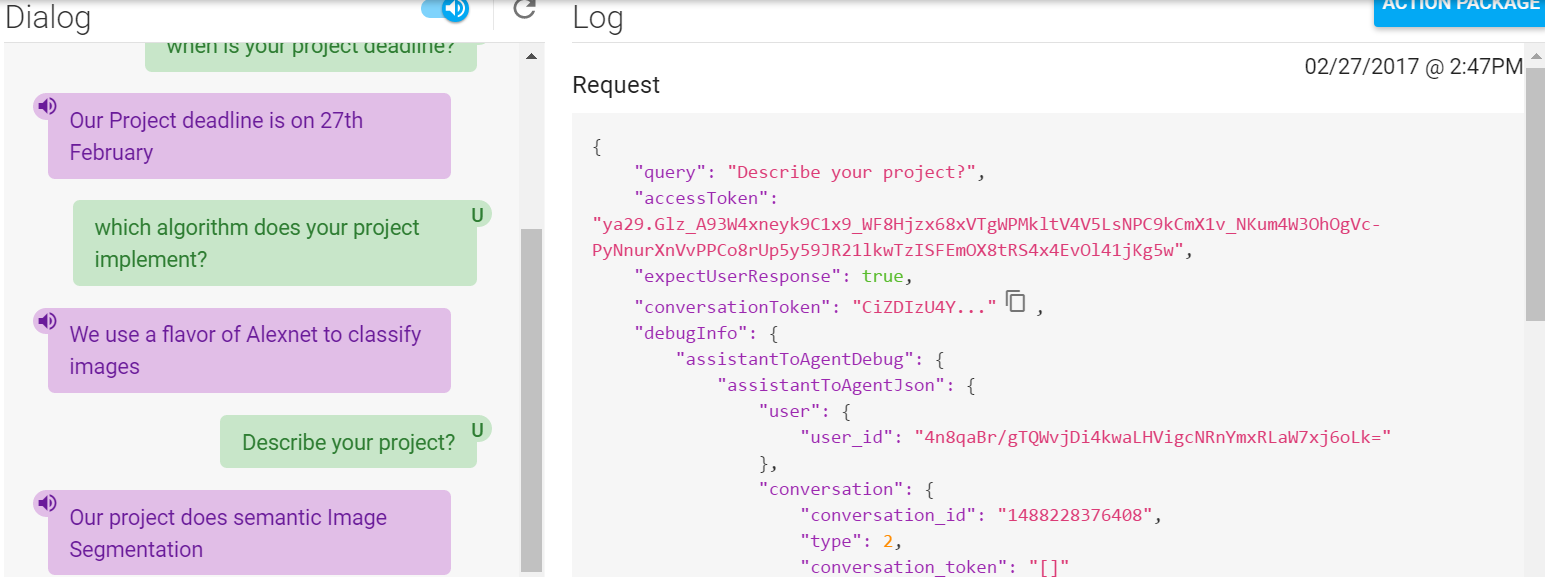


## 5.2.2 Entity Creation

Entities represent concepts and serve as a important tool for extracting parameter values from natural language inputs.

5.2.3 Google Conversation  
Google conversation API enables to have a conversation where the user can ask question the agent answers accordingly.





# 6.Plan & Project Timeline,Members,Task Responsibility

Project plan helps us to schedule the tasks so that everything is finished on time. It helps in time management and allocate work accordingly.

## 6.1 Implementation Status Report

The status report keeps track of the work completed, contribution and the work that yet have to be completed

### 6.1.1 Work Completed

Description: Implemented Google Conversation API,Clarifai API. Collected the data set.

Contribution: Prudhvi : 50%

Nikitha: 50%