PROJECT REPORT

REAL-TIME COMMUNICATION SYSTEM POWERED BY AI SPECIALLY ABLED

submitted by

PNT2022TMID40612

BHUVANESH. K	-	610919104017
ARTHI.A	-	610919104010
KOKILA.M	-	610919104042
JAYASUDHA.K	-	610919104033

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

PROJECT OVERVIEW

Artificial Intelligence is not designed to replace humans but rather to enhance our lives by helping us do things we are unable to do on our own.Many companies are working on this type of research ,including Google deepmind,IBM Watson,Apple sri,Microsoft Cortana, ect.,Which means there will likely be many new developments soon.These innovations could positively impacteveryone's life – even those without disabilities – because the make everyday tasks easier and less time-consuing.

PURPOSE

A making use of a convolution neural to create a model that is trained on different hand gestures.An app is built which uses this model.This app enables deaf and dumb people to convey their information using signs which get converted to human-understandable language and speech is gives as output. Facial recognition technology is quickly becoming a part of everyday life. It's used to improve public security, the accuracy of photo tagging and even make grocery shopping easier. But those who can't speak or move? Facial recognition has the potential to offer independence and inclusion for these individuals. This means that people with disabilities can get a job or go out without needing a caregiver or companion to help them find their way around and do things independently. From entertainment to security, many aspects of daily life have been improved through this advancement in technology. These technologies reached their peak when smartphones became more available to the public market. Today, facial recognition software is being used for blind children to read books aloud and as an accessible way for deaf people to communicate with others via video chat.

CHAPTER 2

LITERATURE SURVEY

EXISTING PROBLEM

The fundamental problem with handwritten digit recognition is that handwritten digits do not always have the same size, width, orientation, and margins since they vary from person to person. Additionally, there would be issues with identifying the numbers because of similarities between numerals like 1 and 7, 5 and 6, 3 and 8, 2 and 5, 2 and 7, etc. Finally, the individuality and variation of each individual's handwriting influence the structure and appearance of the digits.

REFERENCES

Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN) (2020)

Ahlawat, Savita and Choudhary, Amit and Nayyar, Anand and Singh, Saurabh and Yoon, Byungun

This paper's primary goal was to enhance handwritten digit recognition ability. Toavoid difficult pre-processing, expensive feature extraction, and a complex ensemble (classifier combination) method of a standard recognition system, they examined different convolutional neural network variations. Their current work makes suggestions on the function of several hyper-parameters through thorough evaluation utilizing an MNIST dataset. They also confirmed that optimizing

hyper-parameters is crucial for enhancing CNN architecture performance. With the Adam optimizer for the MNIST database, they were able to surpass many previously published results with a recognition rate of 99.89%. Through the trials, it is made abundantly evident how the performance of handwritten digit recognition is affected by the number of convolutional layers in CNN architecture. According to the paper, evolutionary algorithms can be explored for optimizing convolutional filter kernel sizes, CNN learning parameters, and the quantity of layers and learning rates.

An Efficient And Improved Scheme For Handwritten Digit Recognition Based On Convolutional Neural Network (2019)

Ali, Saqib and Shaukat, Zeeshan and Azeem, Muhammad and Sakhawat, Zareen and Mahmood, Tariq and others

This study uses rectified linear units (ReLU) activation and a convolutional neural network (CNN) that incorporates the Deeplearning4j (DL4J) architecture to recognize handwritten digits. The proposed CNN framework has all the necessary parameters for a high level of MNIST digit classification accuracy. The system's training takes into account the time factor as well. The system is also tested by altering the number of CNN layers for additional accuracy verification. It is importantto note that the CNN architecture consists of two convolutional layers, the first with 32 filters and a 5x5 window size and the second with 64 filters and a 7x7 window size. In comparison to earlier proposed systems, the experimental findings show thatthe proposed CNN architecture for the MNIST dataset demonstrates great performance in terms of time and accuracy. As a result, handwritten numbers are detected with a recognition rate of 99.89% and high precision (99.21%) in a short amount of time.

Handwritten Digit Recognition Using Machine And Deep Learning Algorithms (2021)

Pashine, Samay and Dixit, Ritik and Kushwah, Rishika

In this study, they developed three deep and machine learning-based models forhandwritten digit recognition using MNIST datasets. To determine which model was the most accurate, they compared them based on their individual properties.

Support vector machines are among the simplest classifiers, making them faster than other algorithms and providing the highest training accuracy rate in this situation. However, due to their simplicity, SVMs cannot categorize complicated and ambiguous images as accurately as MLP and CNN algorithms can. In their research, they discovered that CNN produced the most precise outcomes for handwritten digit recognition. This led them to the conclusion that CNN is the most effective solution for all types of prediction issues, including those using picture data. Next, by comparing the execution times of the algorithms, they determined that increasing the number of epochs without changing the configuration of the algorithm is pointless due to the limitation of a certain model, and they discovered that beyond a certain number of epochs, the model begins over-fitting the dataset and provides biased predictions.

PROBLEM STATEMENT DEFINITION

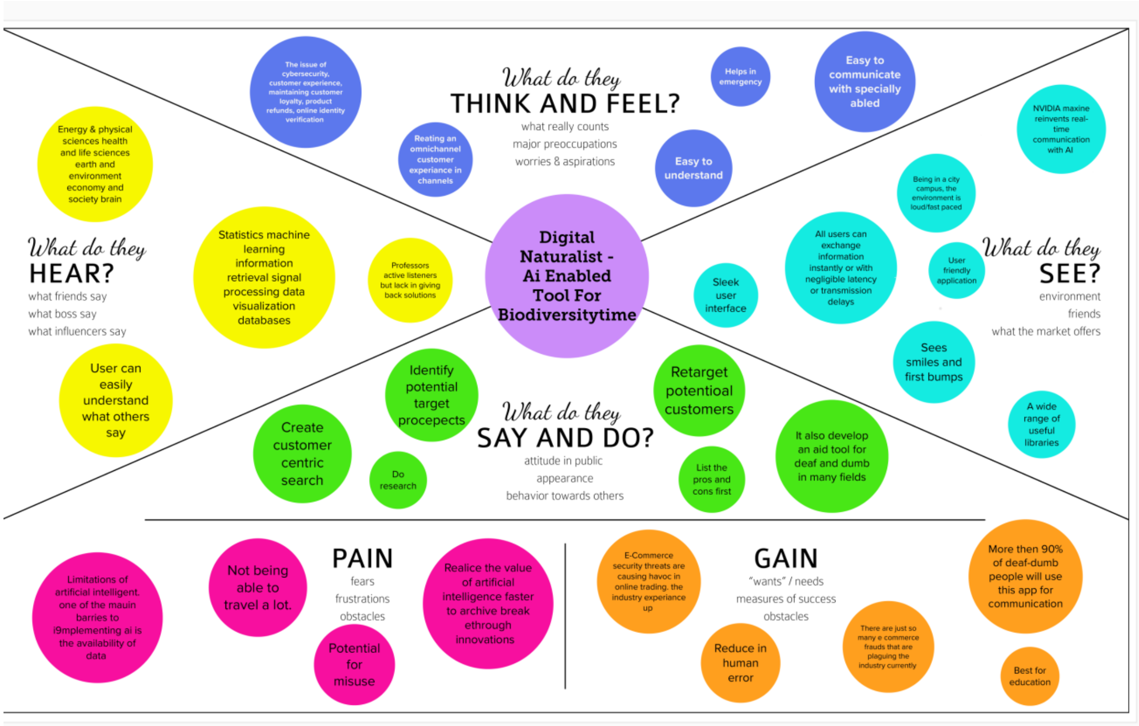
Artificial Intelligence has been opening up new and simpler ways to manage our daily activities. With the big potential to automate tasks that typically require human intelligence, such as speech and voice recognition, visual perception, predictive text functionality, decision-making and performance of a variety of other tasks, AI can help individuals with disabilities by making a major difference in their ability to get around and take part in the activities of daily living.

The Problem for AI Using driverless cars enables disabled people to leave the house, get around their communities, interact with people, and even find jobs. Once autonomous vehicles are fully integrated into society, they could ease independent mobility, and increased accessibility adapted to each user's abilities and needs. Artificial Intelligence has been opening up new and simpler ways to manage our daily activities. With the big potential to automate tasks that typically require human intelligence, such as speech and voice recognition, visual perception, predictive text functionality, decision-making and performance of a variety of other tasks, AI can help individuals with disabilities by making a major difference in their ability to get around and take part in the activities of daily living.

CHAPTER 3

IDEATION AND PROPOSED SOLUTION

EMPATHY MAP CANVAS



Digital Naturalist - Ai Enabled Tool For Biodiversitytime

What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations

The issue of cybersecurity, customer experience, maintaining customer loyalty, product refunds, online identity verification

Reating an omnichannel customer experiance in channels

Helps in emergency

Easy to communicate with specially abled

Easy to understand

What do they HEAR?
what friends say
what boss say
what influencers say

Energy & physical sciences health and life sciences earth and environment economy and society brain

Statistics machine learning information retrieval signal processing data visualization databases

Professors active listeners but lack in giving back solutions

User can easily understand what others say

What do they SEE?
environment
friends
what the market offers

NVIDIA maxine reinvents real-time communication with AI

Being in a city campus, the environment is loud/fast paced

All users can exchange information instantly or with negligible latency or transmission delays

User friendly application

Sleek user interface

Sees smiles and first bumps

A wide range of useful libraries

What do they SAY AND DO?
attitude in public
appearance
behavior towards others

Identify potential target procepects

Create customer centric search

Do research

Retarget potentioal customers

List the pros and cons first

It also develop an aid tool for deaf and dumb in many fields

PAIN
fears
frustrations
obstacles

Limitations of artificial intelligent. one of the mauin barries to i9mplementing ai is the availability of data

Not being able to travel a lot.

Potential for misuse

Realice the value of artificial intelligence faster to archive break ethrough innovations

GAIN
"wants" / needs
measures of success
obstacles

E-Commerce security threats are causing havoc in online trading. the industry experiance up

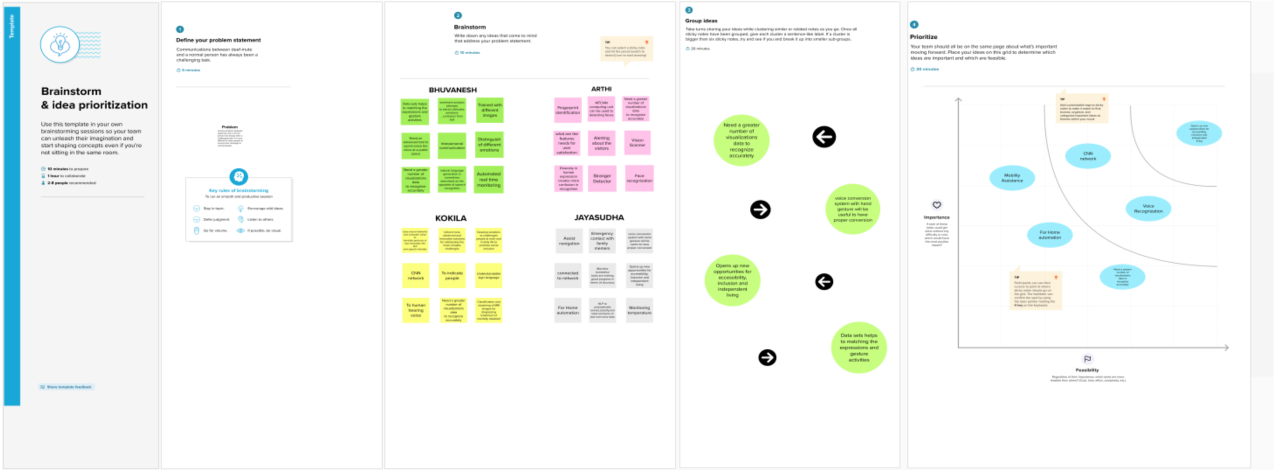
Reduce in human error

There are just so many e commerce frauds that are plaguing the industry currently

More then 90% of deaf-dumb people will use this app for communication

Best for education

IDEATION & BRAINSTORMING



Template

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- **10 minutes** to prepare
- **1 hour** to collaborate
- **2-8 people** recommended

Share template feedback

1

Define your problem statement

Communications between deaf-mute and a normal person has always been a challenging task.

5 minutes

Problem

Key rules of brainstorming

To run an smooth and productive session

- Stay in topic.
- Encourage wild ideas.
- Defer judgment.
- Listen to others.
- Go for volume.
- If possible, be visual.

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

BHUVANESH

- Data sets helps to matching the expressions and gesture activities
- sentiment analysis attempts to extract attitudes, emotions, confusion from text
- Trained with different images
- Need an advanced tool to synchronize the voice at a public space
- Interpersonal communication
- Distinguish of different emotions
- Need a greater number of visualizations data to recognize accurately
- natural language generation is sometimes described as the opposite of speech recognition
- Automated real time monitoring

ARTHI

- Fingerprint identification
- MTCNN computing unit can be used to detecting faces
- Need a greater number of visualizations data to recognize accurately
- what are the features needs for user satisfaction
- Alerting about the visitors
- Vision Scanner
- Diversity in human expression creates more confusion in recognition
- Stranger Detector
- Face recognization

KOKILA

- Uses neural networks and computer vision to translate gestures in Sign language into text and speech instantly
- Unlock more advanced and innovative solutions for addressing the most complex challenges.
- Develop solutions to challenged people at work and in daily life to promote social inclusion
- CNN network
- To indicate people
- Understandable sign language
- To human hearing voice
- Need a greater number of visualizations data to recognize accurately
- Classification and clustering of MRI images for diagnosing treatment of mentally disabled

JAYASUDHA

- Assist navigation
- Emergency contact with family members
- voice conversion system with hand gesture will be useful to have proper conversion
- connected to network
- Machine translation tools are making good progress in terms of accuracy
- Opens up new opportunities for accessibility, inclusion and independent living
- For Home automation
- NLP to automatically extract, classify and label elements of text and voice data
- Monitoring temperature

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

- Need a greater number of visualizations data to recognize accurately
- voice conversion system with hand gesture will be useful to have proper conversion
- Opens up new opportunities for accessibility, inclusion and independent living
- Data sets helps to matching the expressions and gesture activities

4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

TIP

Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.

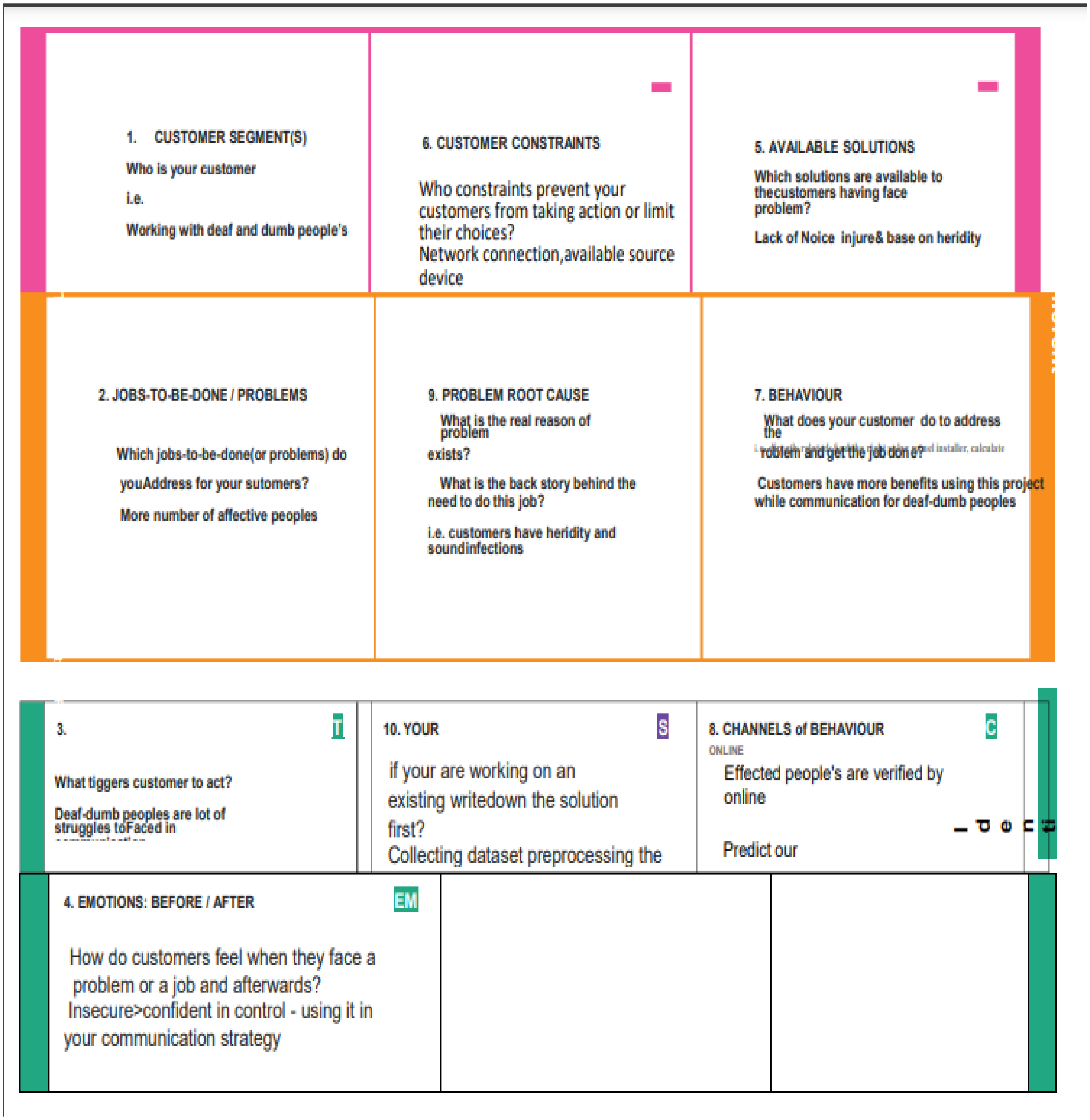
Opens up new opportunities for accessibility, inclusion and independent living
CNN network
Mobility Assistance
Voice Recognization
For Home automation
Need a greater number of visualizations data to recognize accurately
Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

PROPOSED SOLUTION

S.NO	PARAMETER	DESCRIPTION
1	Problem Statement	The main objectives is to build a communiation system which enables communication between a speech hearing impaired and a normal person
2	Idea / Solution Description	The proposed solution uses a Deep Neural Network architecture that recognizes a sign language symbol.The image of the symbol or sign mode by a person is captured via a webcam,which is then fed into the model.
3	Novelty / Uniqueness	The proposed model is more efficient and can also be accessible by lots of people since it will be deployed on the internet with a user-friendly interface
4	Social Impact / CustomerSatisfaction	This model introduced the gateway for deaf, and the blind. It’s difficult to educate the public about the language of disabled people and this model will actually make communication easier and bridge the gap between people
5	Business Model	This model will be made easily accessible to the general public and satisfies their exiting needs and also provides for their new needs. The cost will be user friendly, with different updates,cost may vary

6	Scalability of the Solution	With adequate funding and manpower,the proposed model can be scaled up,which would make it a more sophisticate system that can recognize multipile sign languages and also convert into multiple normal languages

PROBLEM SOLUTION FIT



1. CUSTOMER SEGMENT(S)

Who is your customer

i.e.

Working with deaf and dumb people's

6. CUSTOMER CONSTRAINTS

Who constraints prevent your customers from taking action or limit their choices?
Network connection,available source device

5. AVAILABLE SOLUTIONS

Which solutions are available to thecustomers having face problem?

Lack of Noice injure& base on heridity

2. JOBS-TO-BE-DONE / PROBLEMS

Which jobs-to-be-done(or problems) do youAddress for your sutomers?

More number of affective peoples

9. PROBLEM ROOT CAUSE

What is the real reason of problem exists?

What is the back story behind the need to do this job?

i.e. customers have heridity and soundinfections

7. BEHAVIOUR

What does your customer do to address the roblem and get the job done?

Customers have more benefits using this project while communication for deaf-dumb peoples

3. T

What tiggers customer to act?

Deaf-dumb peoples are lot of struggles toFaced in

10. YOUR S

if your are working on an existing writedown the solution first?
Collecting dataset preprocessing the

8. CHANNELS of BEHAVIOUR C

ONLINE

Effected people's are verified by online

Predict our

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
Insecure>confident in control - using it in your communication strategy

CHAPTER 4

REQUIREMENT ANALYSIS

FUNCTIONAL REQUIREMENTS:

FR.NO	FUNCTIONAL REQUIREMENTS	SUB REQUIREMENTS
FR-1	Model Creation	Get access the MNIST dataset
		Analyze the dataset
		Define a CNN model
		Train and Test the Model
FR-2	Application Development	Create a website to let the user recognize handwritten digits.
		Create a home page to upload images
		Create a result page to display the results
		Host the website to let the users use it from anywhere
FR-3	Input Image Upload	Let users upload images of various formats.
		Let users upload images of various size
		Prevent users from uploading unsupported image formats
		Pre-Process the image to use it on the model

		Create a database to store all the input images
FR-4	Display Results	Display the result from the model
		Display input image
		Display accuracy the result
		Display other possible predictions with their respective accuracy

NON FUNCTIONAL REQUIREMENTS

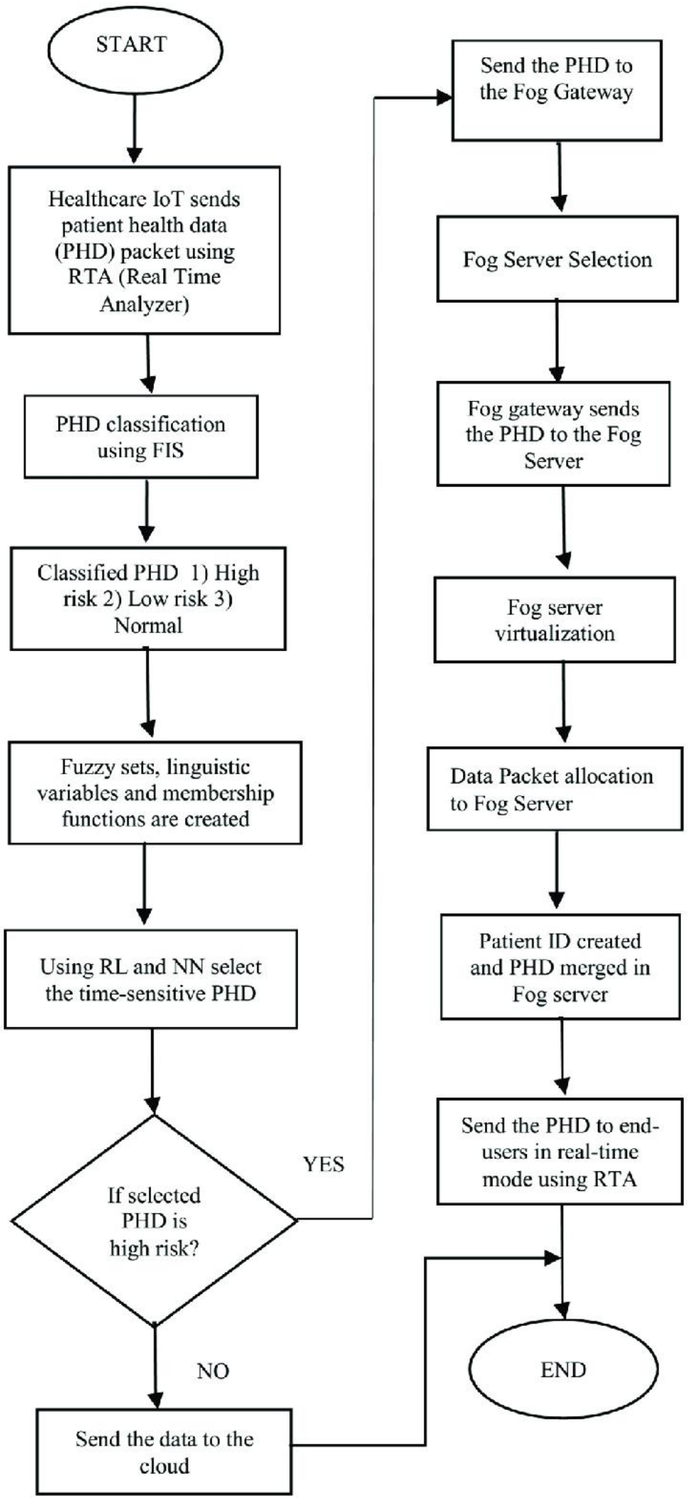
NFR	NON-FUNCTIONAL REQUIREMENTS	DESCRIPTION
NFR-1	Usability	The application must be usable in all devices
NFR-2	Security	The application must protect user uploaded image
NFR-3	Reliability	The application must give an accurate result as much as possible

NFR-4	Performance	The application must be fast and quick to load up
NFR-5	Availability	The application must be available to use all the time
NFR-6	Scalability	The application must scale along with the user base

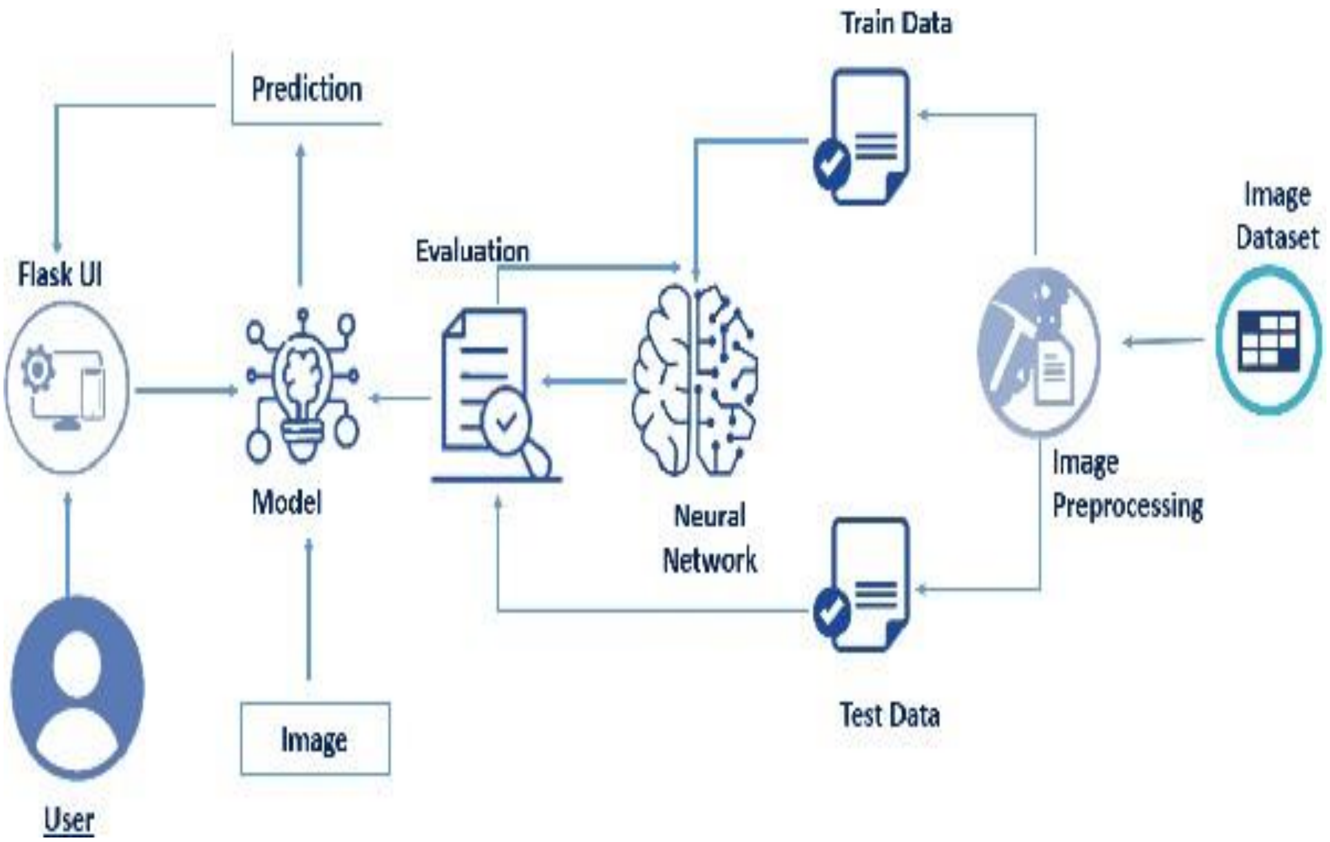
CHAPTER 5

PROJECT DESIGN

DATA FLOW DIAGRAM


START
Healthcare IoT sends patient health data (PHD) packet using RTA (Real Time Analyzer)
PHD classification using FIS
Classified PHD 1) High risk 2) Low risk 3) Normal
Fuzzy sets, linguistic variables and membership functions are created
Using RL and NN select the time-sensitive PHD
If selected PHD is high risk?
YES
NO
Send the data to the cloud
Send the PHD to the Fog Gateway
Fog Server Selection
Fog gateway sends the PHD to the Fog Server
Fog server virtualization
Data Packet allocation to Fog Server
Patient ID created and PHD merged in Fog server
Send the PHD to end-users in real-time mode using RTA
END

SOLUTION & TECHNICAL ARCHITECTURE



USER STORIES

User Type	Functional Requirement(Epic)	User Story Number	User Story/ Task	Acceptanc e criteria	Priority	Release
Normal people and Deaf-mute people	Registration	USN-1	As a user, I can register for the application by entering my email, and password, and confirming my password	I can access my account /d ashboard	High	Sprint-1
		USN-2	As a user, I will receive a confirmation email once I have registered for the application	I can receive a confirmati on email & click confirm	High	Sprint-1
Normal people		USN-3	Give access tocamera torecognize the gestures Give access tomicrophone to give our message through voice	I can access messag esgiven by the Deaf-mute people	High	Sprint-1
Deaf-mute people			Give access to display toview the message sent by normal people.	I can access messag esgiven by the Normal people	High	Sprint-1
Administrator		USN-4	Admin side in the company should takecare.	all the require me nts are	High	Sprint-1

				there		
Sign up		USN-5	Need to sign up to use it.	Need valid credentials	High	Sprint-1
Wish list		USN-6	Before availing the service can be kept aside.	As a user can review and use the service.	Low	Sprint-2

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

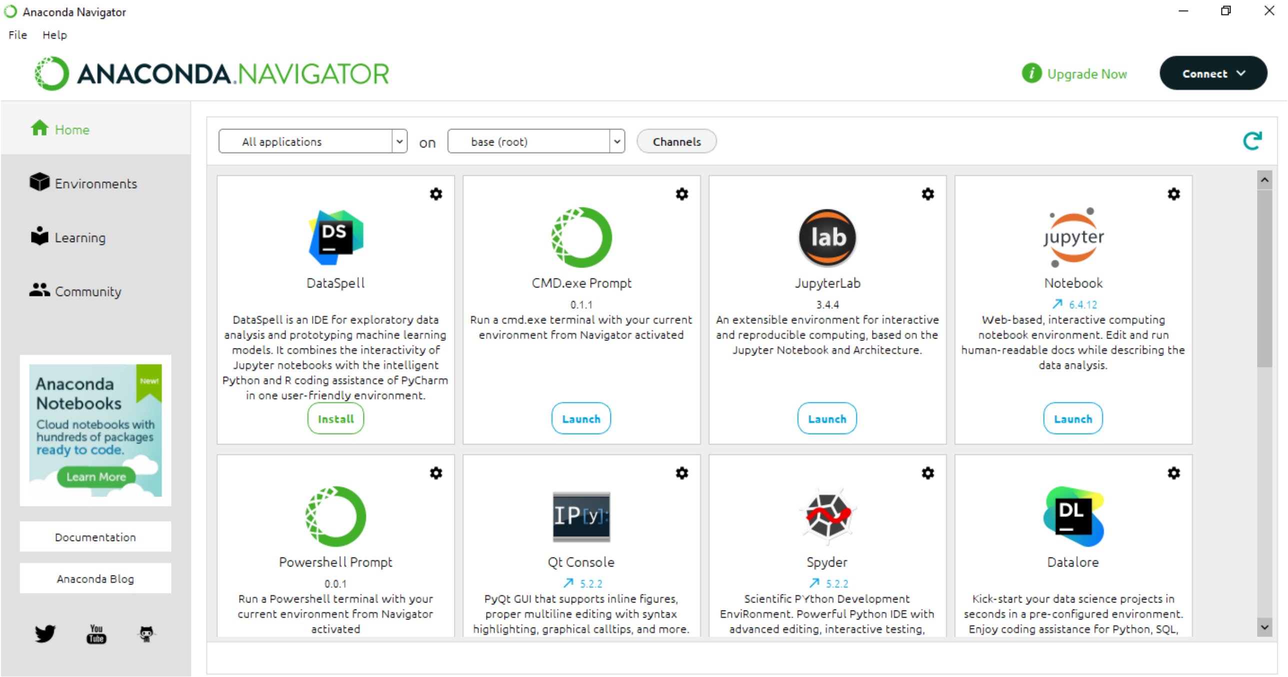
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	BHUVANESH K
Sprint-2	Action	USN-2	As a user, I will receive confirmation email onceI have registered for the application	1	High	KOKILA M
Sprint-1	Login	USN-3	As a user, I can log into the application by entering email & password	1	Medium	BHUVANESH K
Sprint-2	Dashboard	USN-4	As a user, I can log into my account in a givenDashboard	1	High	KOKILA M
Sprint-1	User interface	USN-5	Professional responsible for user requirements &needs	1	High	BHUVANESH K
Sprint-3	Objective	USN-6	The goal is to describe all the inputs and outputs	1	High	KOKILA M
Sprint-4	Privacy	USN-7	The developed application should be secure forthe users	1	High	JAYASUDHA K

SPRINT DELIVERY SCHEDULE

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	6 Nov 2022	16 Nov 2022	20	16 Nov 2022
Sprint-2	20	6 Days	11 Nov 2022	17 Nov 2022	20	17 Nov 2022
Sprint-3	20	6 Days	13 Nov 2022	19 Nov 2022	20	19 Nov 2022
Sprint-4	20	6 Days	15 Nov 2022	20 Nov 2022	20	20 Nov 2022

CHAPTER 7

CODING & SOLUTIONING



DataSpell

DataSpell is an IDE for exploratory data analysis and prototyping machine learning models. It combines the interactivity of Jupyter notebooks with the intelligent Python and R coding assistance of PyCharm in one user-friendly environment.

CMD.exe Prompt

0.1.1

Run a cmd.exe terminal with your current environment from Navigator activated

JupyterLab

3.4.4

An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook

6.4.12

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Powershell Prompt

0.0.1

Run a Powershell terminal with your current environment from Navigator activated

Qt Console

5.2.2

PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

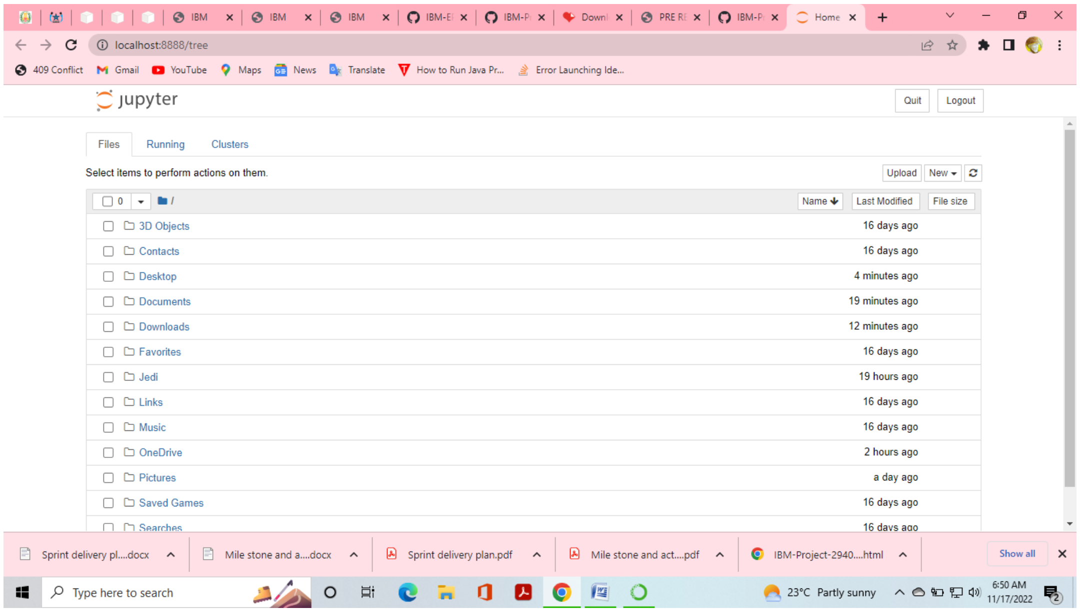
Spyder

5.2.2

Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing,

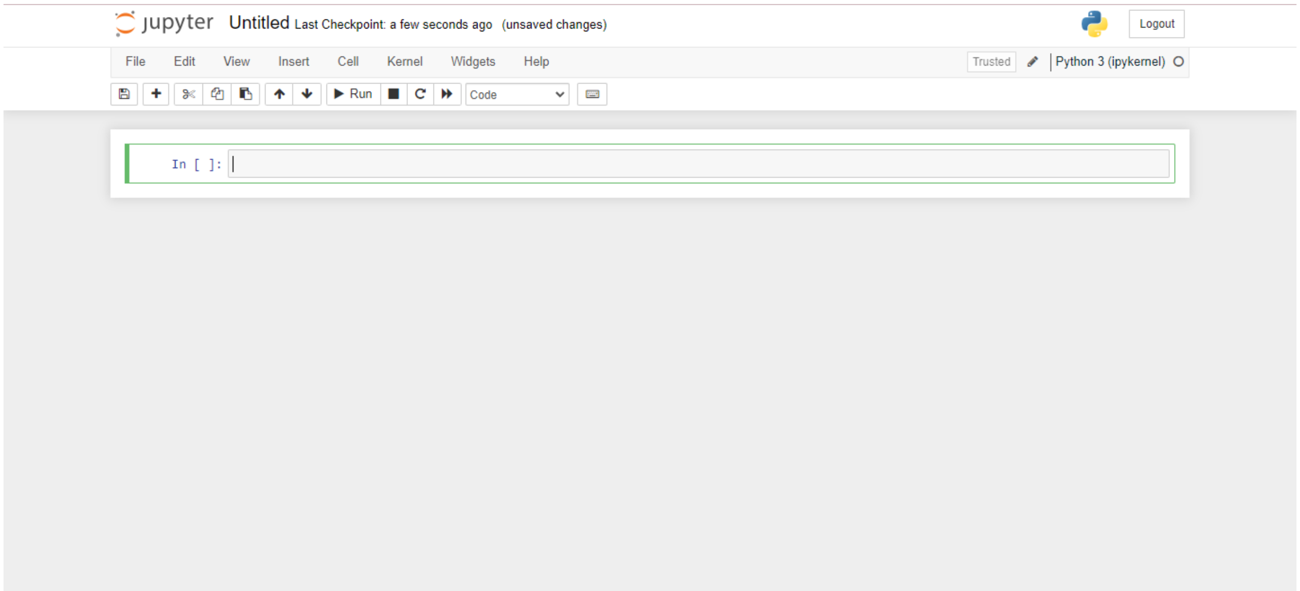
Datalore

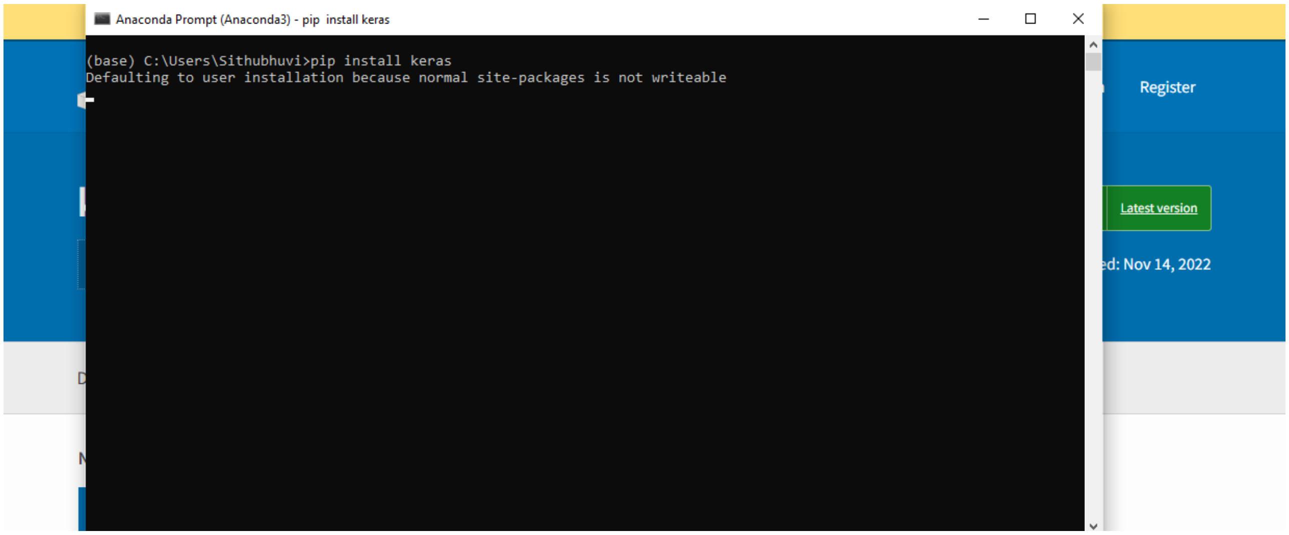
Kick-start your data science projects in seconds in a pre-configured environment. Enjoy coding assistance for Python, SQL,



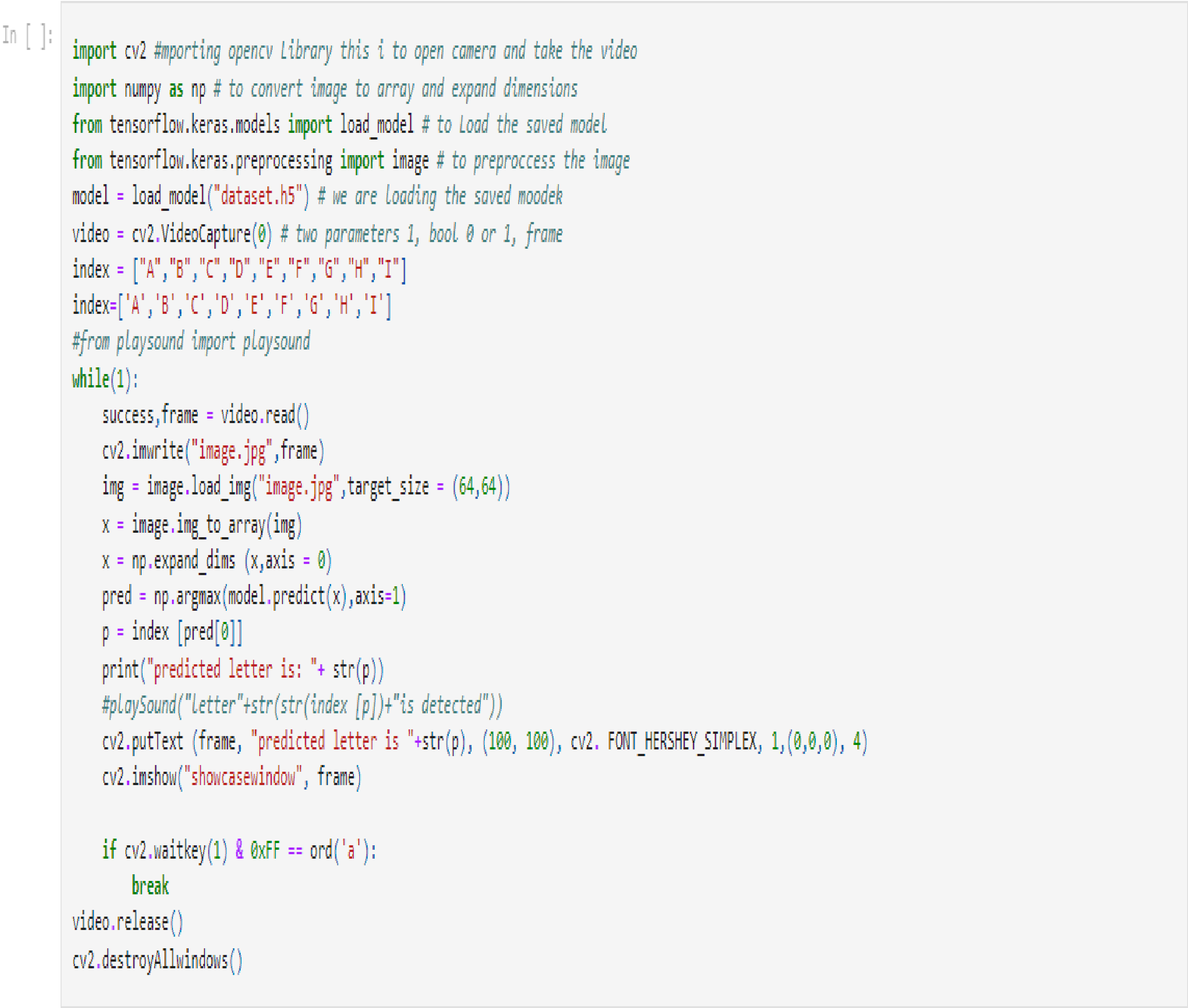
Select items to perform actions on them.

Name	Last Modified
3D Objects	16 days ago
Contacts	16 days ago
Desktop	4 minutes ago
Documents	19 minutes ago
Downloads	12 minutes ago
Favorites	16 days ago
Jedi	19 hours ago
Links	16 days ago
Music	16 days ago
OneDrive	2 hours ago
Pictures	a day ago
Saved Games	16 days ago
Searches	16 days ago





```
(base) C:\Users\Sithubhuvi>pip install keras
Defaulting to user installation because normal site-packages is not writeable
```



In [ ]:

```
import cv2 #mporting opencv library this i to open camera and take the video
import numpy as np # to convert image to array and expand dimensions
from tensorflow.keras.models import load_model # to Load the saved model
from tensorflow.keras.preprocessing import image # to preproccess the image
model = load_model("dataset.h5") # we are loading the saved moodek
video = cv2.VideoCapture(0) # two parameters 1, bool 0 or 1, frame
index = ["A","B","C","D","E","F","G","H","I"]
index=['A','B','C','D','E','F','G','H','I']
#from playsound import playsound
while(1):
    success,frame = video.read()
    cv2.imwrite("image.jpg",frame)
    img = image.load_img("image.jpg",target_size = (64,64))
    x = image.img_to_array(img)
    x = np.expand_dims (x,axis = 0)
    pred = np.argmax(model.predict(x),axis=1)
    p = index [pred[0]]
    print("predicted letter is: "+ str(p))
    #playSound("letter"+str(str(index [p])+"is detected"))
    cv2.putText (frame, "predicted letter is "+str(p), (100, 100), cv2. FONT_HERSHEY_SIMPLEX, 1,(0,0,0), 4)
    cv2.imshow("showcasewindow", frame)

    if cv2.waitkey(1) & 0xFF == ord('a'):
        break
video.release()
cv2.destroyAllwindows()
```

CHAPTER 8
TESTING

TEST CASES

Test case ID	Feature Type	Component	Test Scenario	Expected Result	Actual Result	Status
HP_TC_001	UI	Home Page	Verify UI elements in the Home Page	The Home page must be displayed properly	Working as expected	PASS
HP_TC_002	UI	Home Page	Check if the UI elements are displayed properly in different screen sizes	The Home page must be displayed properly in all sizes	The UI is not displayed properly in screen size 2560 x 1801 and 768 x 630	FAIL
HP_TC_003	Functiona	Home Page	Check if user can upload their file	The input image should be uploaded to the application successfully	Working as expected	PASS
HP_TC_004	Functiona	Home Page	Check if user cannot upload unsupported files	The application should not allow user to select a non image file	User is able to upload any file	FAIL

HP_TC_005	Functiona	Home Page	Check if the page redirects to the result page once the input is given	The page should redirect to the results page	Working as expected	PASS

BE_TC_001	Functiona	Backend	Check if all the routes are working properly	All the routes should properly work	Working as expected	PASS
M_TC_001	Functiona	Model	Check if the model can handle various image sizes	The model should rescale the image and predict the results	Working as expected	PASS
M_TC_002	Functiona	Model	Check if the model predicts the digit	The model should predict the number	Working as expected	PASS
M_TC_003	Functiona	Model	Check if the model can handle complex input image	The model should predict the number in the complex image	The model fails to identify the digit since the model is not built to handle such data	FAIL
RP_TC_001	UI	Result Page	Verify UI elements in the Result Page	The Result page must be displayed properly	Working as expected	PASS
RP_TC_002	UI	Result Page	Check if the input image is displayed properly	The input image should be displayed properly	The size of the input image exceeds the display container	FAIL
RP_TC_003	UI	Result Page	Check if the result is displayed properly	The result should be displayed properly	Working as expected	PASS
RP_TC_004	UI	Result Page	Check if the other predictions are displayed properly	The other predictions should be displayed properly	Working as expected	PASS

USER ACCEPTANCE TESTING

DEFECT ANALYSIS

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Total
By Design	1	0	1	0	2
Duplicate	0	0	0	0	0
External	0	0	2	0	2
Fixed	4	1	0	1	6
Not Reproduced	0	0	0	1	1
Skipped	0	0	0	1	1
Won’t Fix	1	0	1	0	2
Total	6	1	4	3	14

TEST CASE ANALYSIS

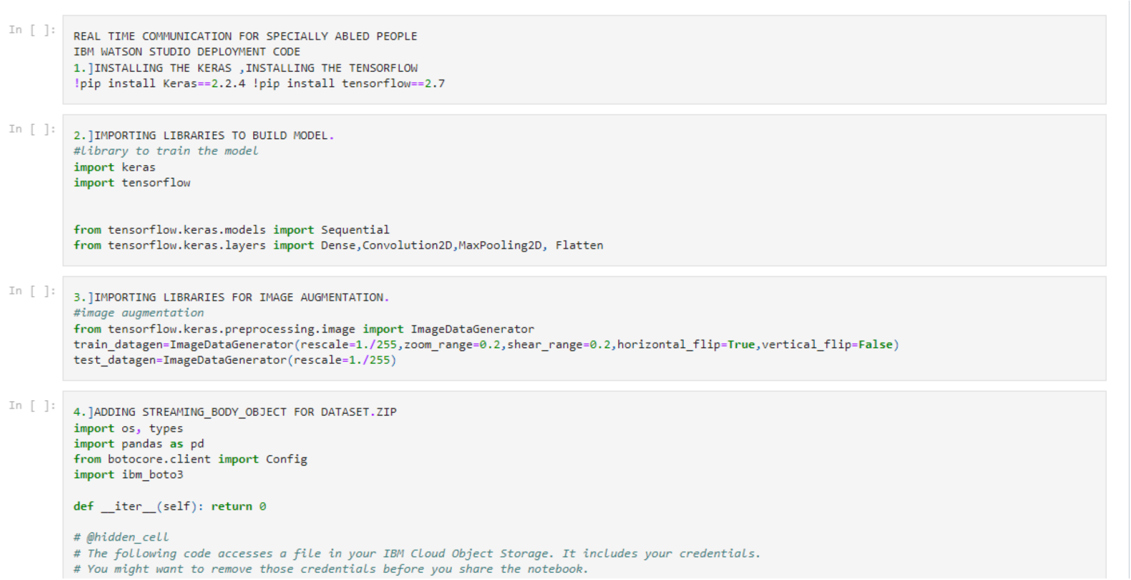
Section	Total Cases	Not Tested	Fail	Pass
Client Application	10	0	3	7
Security	2	0	1	1

Performance	3	0	1	2
Exception Reporting	2	0	0	2

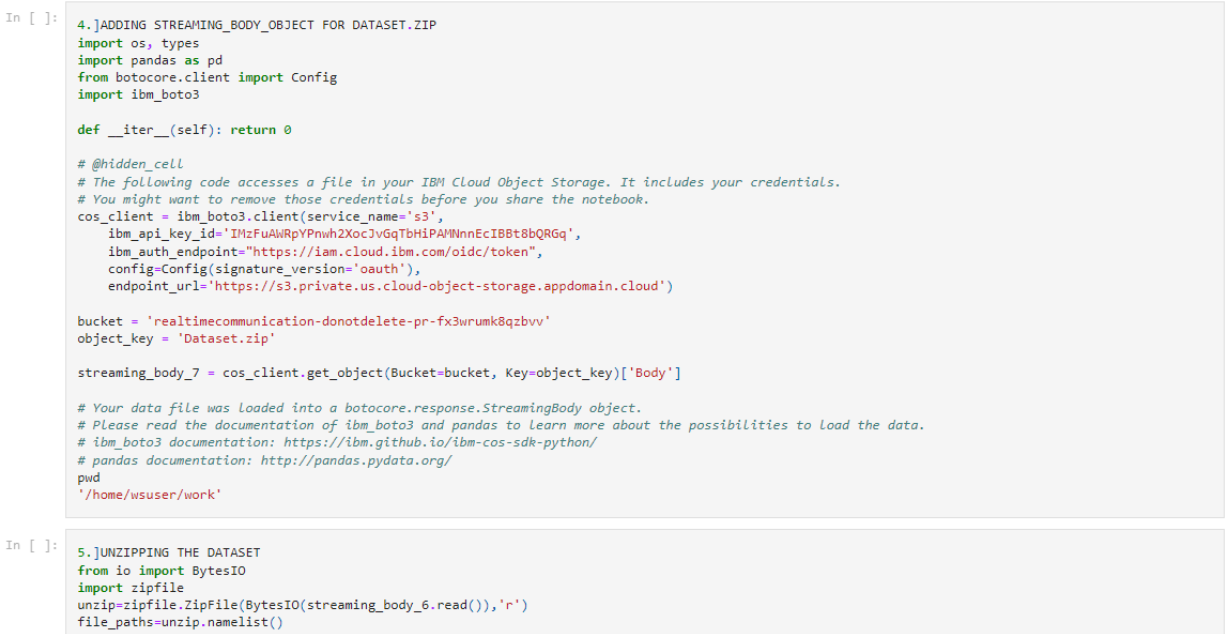
CHAPTER 9

RESULTS

PERFORMANCE METRICS

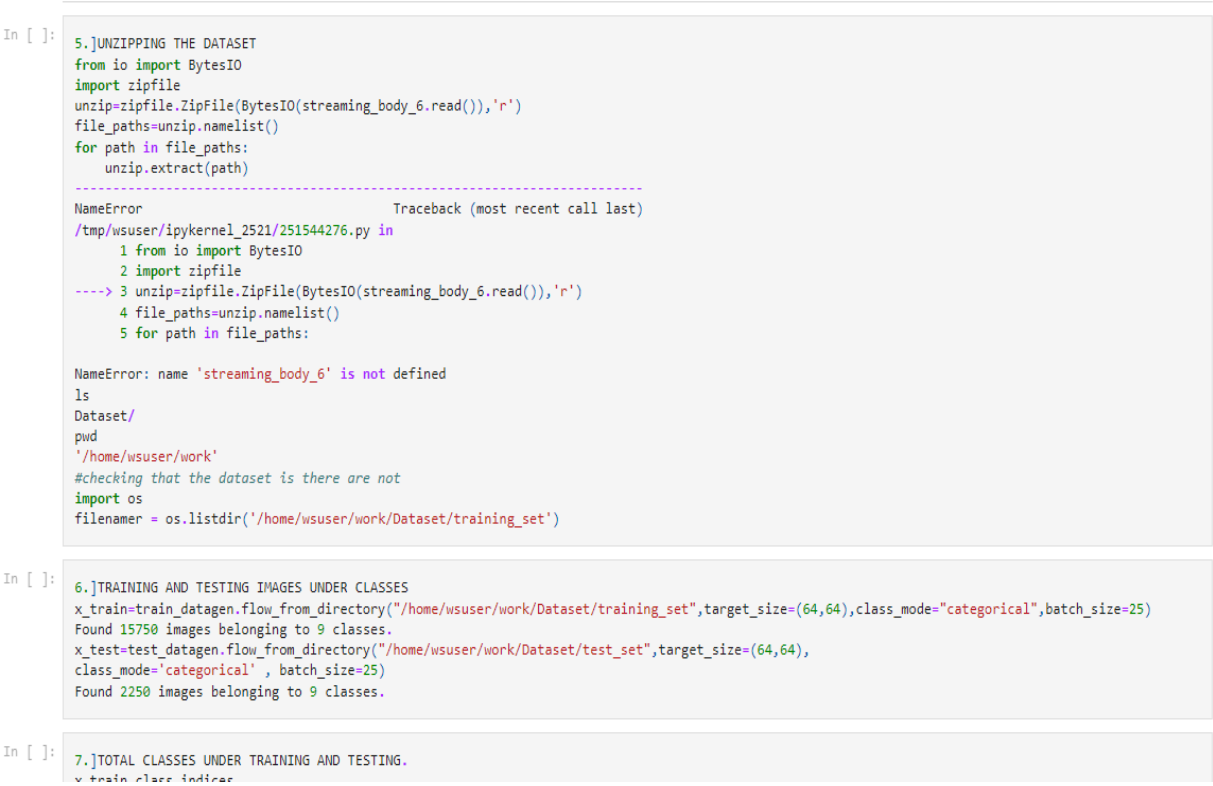


```
In [ ]:
REAL TIME COMMUNICATION FOR SPECIALLY ABLED PEOPLE
IBM WATSON STUDIO DEPLOYMENT CODE
1.]INSTALLING THE KERAS ,INSTALLING THE TENSORFLOW
!pip install Keras==2.2.4 !pip install tensorflow==2.7

In [ ]:
2.]IMPORTING LIBRARIES TO BUILD MODEL.
#library to train the model
import keras
import tensorflow


from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D, Flatten

In [ ]:
3.]IMPORTING LIBRARIES FOR IMAGE AUGMENTATION.
#image augmentation
from tensorflow.keras.preprocessing.image import ImageDataGenerator
train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,shear_range=0.2,horizontal_flip=True,vertical_flip=False)
test_datagen=ImageDataGenerator(rescale=1./255)

In [ ]:
4.]ADDING STREAMING_BODY_OBJECT FOR DATASET.ZIP
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
```



```
In [ ]:
4.]ADDING STREAMING_BODY_OBJECT FOR DATASET.ZIP
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='IMzFuAWRpYPnwh2XocJvGqTbHiPAMNnnEcIBBt8bQRGq',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'realtimecommunication-donotdelete-pr-fx3wrumk8qzbvv'
object_key = 'Dataset.zip'

streaming_body_7 = cos_client.get_object(Bucket=bucket, Key=object_key)['Body']

# Your data file was loaded into a botocore.response.StreamingBody object.
# Please read the documentation of ibm_boto3 and pandas to learn more about the possibilities to load the data.
# ibm_boto3 documentation: https://ibm.github.io/ibm-cos-sdk-python/
# pandas documentation: http://pandas.pydata.org/
pwd
'/home/wsuser/work'
```

```
In [ ]:
5.]UNZIPPING THE DATASET
from io import BytesIO
import zipfile
unzip=zipfile.ZipFile(BytesIO(streaming_body_6.read()),'r')
file_paths=unzip.namelist()
```



```
In [ ]:
5.]UNZIPPING THE DATASET
from io import BytesIO
import zipfile
unzip=zipfile.ZipFile(BytesIO(streaming_body_6.read()),'r')
file_paths=unzip.namelist()
for path in file_paths:
    unzip.extract(path)
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
/tmp/wsuser/ipykernel_2521/251544276.py in
      1 from io import BytesIO
      2 import zipfile
----> 3 unzip=zipfile.ZipFile(BytesIO(streaming_body_6.read()),'r')
      4 file_paths=unzip.namelist()
      5 for path in file_paths:

NameError: name 'streaming_body_6' is not defined
ls
Dataset/
pwd
'/home/wsuser/work'
#checking that the dataset is there are not
import os
filenamer = os.listdir('/home/wsuser/work/Dataset/training_set')
```

```
In [ ]:
6.]TRAINING AND TESTING IMAGES UNDER CLASSES
x_train=train_datagen.flow_from_directory("/home/wsuser/work/Dataset/training_set",target_size=(64,64),class_mode="categorical",batch_size=25)
Found 15750 images belonging to 9 classes.
x_test=test_datagen.flow_from_directory("/home/wsuser/work/Dataset/test_set",target_size=(64,64),
class_mode='categorical' , batch_size=25)
Found 2250 images belonging to 9 classes.
```

```
In [ ]:
7.]TOTAL CLASSES UNDER TRAINING AND TESTING.
x_train.class_indices
```

CHAPTER 10

ADVANTAGES & DISADVANTAGES

ADVANTAGES

- Reduces manual work
- More accurate than average human
- Capable of handling a lot of data
- Can be used anywhere from any device

DISADVANTAGES

- Cannot handle complex data
- All the data must be in digital format
- Requires a high performance server for faster predictions
- Prone to occasional errors

CHAPTER 11

CONCLUSION

This project demonstrated a web application that uses machine learning And NuralNetwork torecognise handwritten numbers. Flask, HTML, CSS, JavaScript, and a few other technologies were used to create this project. The model predicts the handwritten digit using a CNN network. During testing, the model achieved a 99.61% recognition rate. The proposed project is scalable and can easily handle a huge number of users.Since it is a web application, it is compatible with any device that can run a browser.This project is extremely useful in real-world scenarios such as recognizing number plates of vehicles, processing bank cheque amounts, numeric entries in forms filledup by hand (tax forms) and so on. There is so much room for improvement, which can be implemented in subsequent versions.

CHAPTER 12

FUTURE SCOPE

This project is far from complete and there is a lot of room for improvement.

Some of the improvements that can be made to this project are as follows:

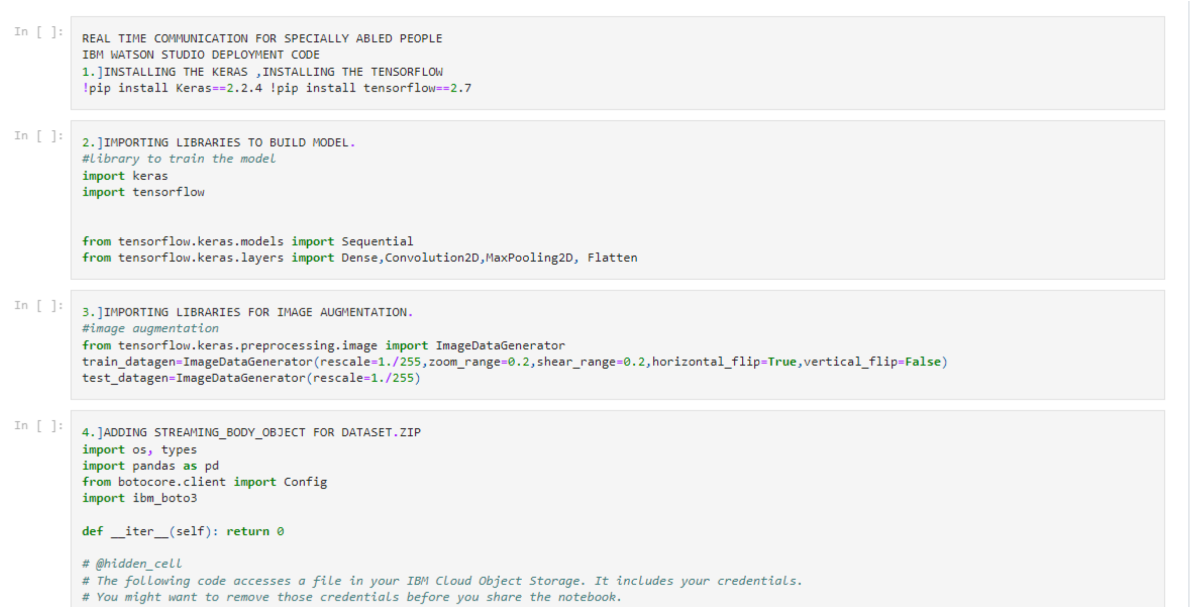
- Add support to detect from digits multiple images and save the results
- Add support to detect multiple digits
- Add support to multi reactions fuction
- Improve model to detect digits from complex images
- Add support to different languages to help users from all over the world

This project has endless potential and can always be enhanced to become better.Implementing this concept in the real world will benefit several industries and reduce the workload on many workers, enhancing overall work efficiency.

APPENDIX

SOURCE CODE

MODEL CREATION

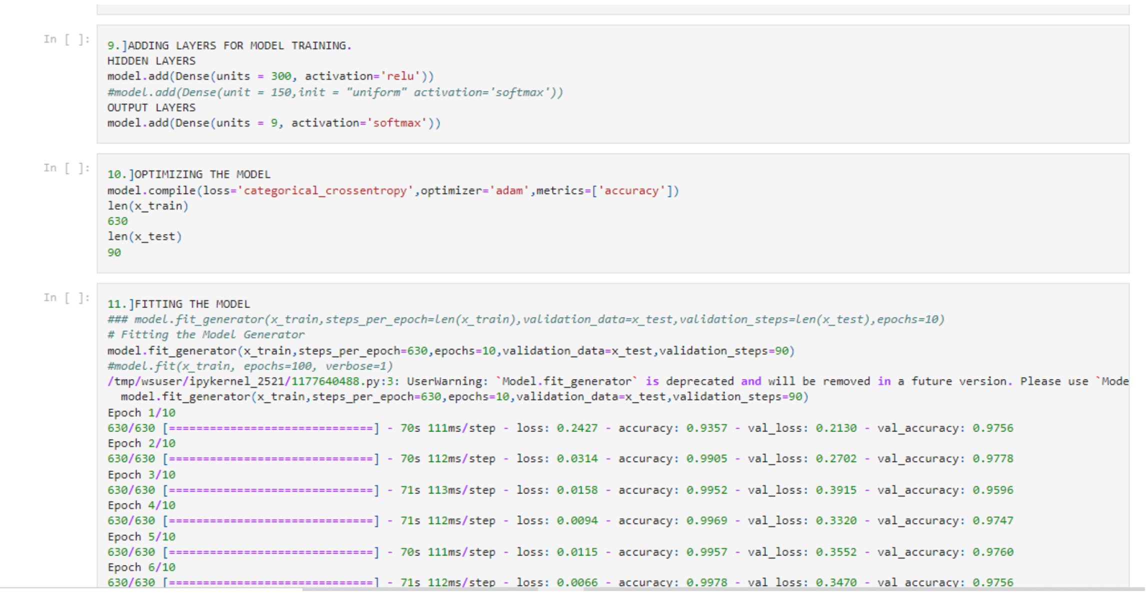


```
In [ ]:
REAL TIME COMMUNICATION FOR SPECIALLY ABLED PEOPLE
IBM WATSON STUDIO DEPLOYMENT CODE
1.]INSTALLING THE KERAS ,INSTALLING THE TENSORFLOW
!pip install Keras==2.2.4 !pip install tensorflow==2.7
```

```
In [ ]:
2.]IMPORTING LIBRARIES TO BUILD MODEL.
#library to train the model
import keras
import tensorflow


from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D, Flatten
```

```
In [ ]:
3.]IMPORTING LIBRARIES FOR IMAGE AUGMENTATION.
#image augmentation
from tensorflow.keras.preprocessing.image import ImageDataGenerator
train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,shear_range=0.2,horizontal_flip=True,vertical_flip=False)
test_datagen=ImageDataGenerator(rescale=1./255)
```

```
In [ ]:
4.]ADDING STREAMING_BODY_OBJECT FOR DATASET.ZIP
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
```



```
In [ ]:
9.]ADDING LAYERS FOR MODEL TRAINING.
HIDDEN LAYERS
model.add(Dense(units = 300, activation='relu'))
#model.add(Dense(unit = 150,init = "uniform" activation='softmax'))
OUTPUT LAYERS
model.add(Dense(units = 9, activation='softmax'))
```

```
In [ ]:
10.]OPTIMIZING THE MODEL
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
len(x_train)
630
len(x_test)
90
```

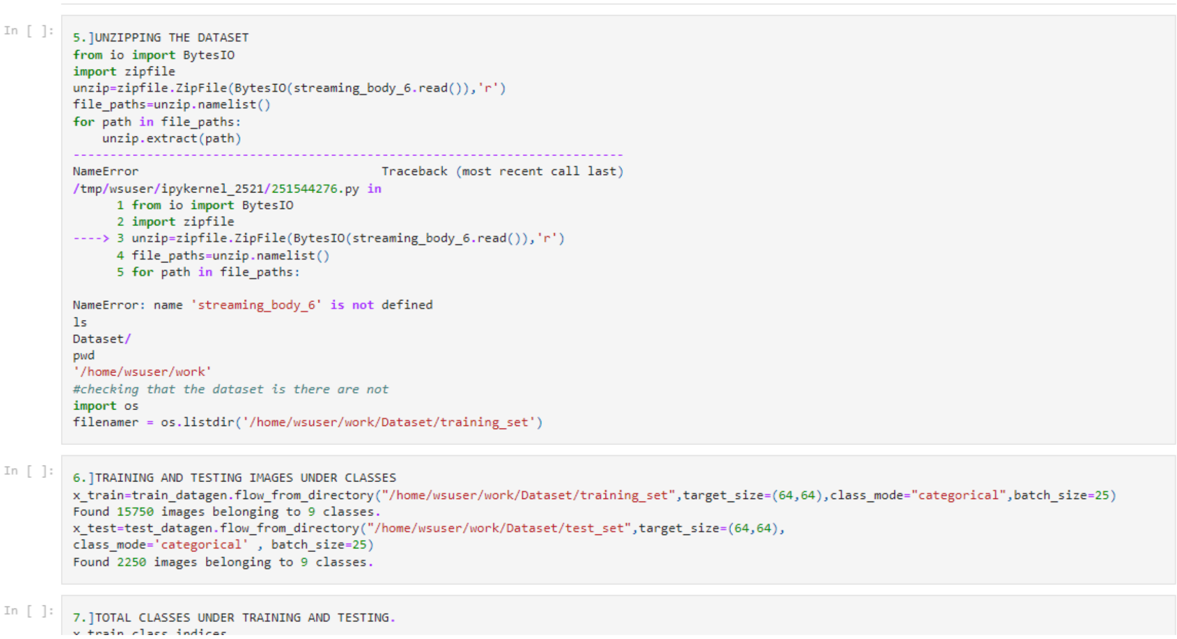
```
In [ ]:
11.]FITTING THE MODEL
### model.fit_generator(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test),epochs=10)
# Fitting the Model Generator
model.fit_generator(x_train,steps_per_epoch=630,epochs=10,validation_data=x_test,validation_steps=90)
#model.fit(x_train, epochs=100, verbose=1)
/tmp/wsuser/ipykernel_2521/1177640488.py:3: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Mode
  model.fit_generator(x_train,steps_per_epoch=630,epochs=10,validation_data=x_test,validation_steps=90)
Epoch 1/10
630/630 [==============================] - 70s 111ms/step - loss: 0.2427 - accuracy: 0.9357 - val_loss: 0.2130 - val_accuracy: 0.9756
Epoch 2/10
630/630 [==============================] - 70s 112ms/step - loss: 0.0314 - accuracy: 0.9905 - val_loss: 0.2702 - val_accuracy: 0.9778
Epoch 3/10
630/630 [==============================] - 71s 113ms/step - loss: 0.0158 - accuracy: 0.9952 - val_loss: 0.3915 - val_accuracy: 0.9596
Epoch 4/10
630/630 [==============================] - 71s 112ms/step - loss: 0.0094 - accuracy: 0.9969 - val_loss: 0.3320 - val_accuracy: 0.9747
Epoch 5/10
630/630 [==============================] - 70s 111ms/step - loss: 0.0115 - accuracy: 0.9957 - val_loss: 0.3552 - val_accuracy: 0.9760
Epoch 6/10
630/630 [==============================] - 71s 112ms/step - loss: 0.0066 - accuracy: 0.9978 - val_loss: 0.3470 - val_accuracy: 0.9756
```

In [ ]:

```
4.]ADDING STREAMING_BODY_OBJECT FOR DATASET.ZIP
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='IMzFuAWRpYPnwh2XocJvGqTbHiPAMNnnEcIBBt8bQRGq',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'realtimecommunication-donotdelete-pr-fx3wrumk8qzbvv'
object_key = 'Dataset.zip'

streaming_body_7 = cos_client.get_object(Bucket=bucket, Key=object_key)['Body']

# Your data file was loaded into a botocore.response.StreamingBody object.
# Please read the documentation of ibm_boto3 and pandas to learn more about the possibilities to load the data.
# ibm_boto3 documentation: https://ibm.github.io/ibm-cos-sdk-python/
# pandas documentation: http://pandas.pydata.org/
pwd
'/home/wsuser/work'
```

In [ ]:

```
5.]UNZIPPING THE DATASET
from io import BytesIO
import zipfile
unzip=zipfile.ZipFile(BytesIO(streaming_body_6.read()),'r')
file_paths=unzip.namelist()
```



In [ ]:

```
5.]UNZIPPING THE DATASET
from io import BytesIO
import zipfile
unzip=zipfile.ZipFile(BytesIO(streaming_body_6.read()),'r')
file_paths=unzip.namelist()
for path in file_paths:
    unzip.extract(path)
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
/tmp/wsuser/ipykernel_2521/251544276.py in
      1 from io import BytesIO
      2 import zipfile
----> 3 unzip=zipfile.ZipFile(BytesIO(streaming_body_6.read()),'r')
      4 file_paths=unzip.namelist()
      5 for path in file_paths:

NameError: name 'streaming_body_6' is not defined
ls
Dataset/
pwd
'/home/wsuser/work'
#checking that the dataset is there are not
import os
filenamer = os.listdir('/home/wsuser/work/Dataset/training_set')
```

In [ ]:

```
6.]TRAINING AND TESTING IMAGES UNDER CLASSES
x_train=train_datagen.flow_from_directory("/home/wsuser/work/Dataset/training_set",target_size=(64,64),class_mode="categorical",batch_size=25)
Found 15750 images belonging to 9 classes.
x_test=test_datagen.flow_from_directory("/home/wsuser/work/Dataset/test_set",target_size=(64,64),
class_mode='categorical' , batch_size=25)
Found 2250 images belonging to 9 classes.
```

In [ ]:

```
7.]TOTAL CLASSES UNDER TRAINING AND TESTING.
x_train.class_indices
```

In [ ]:

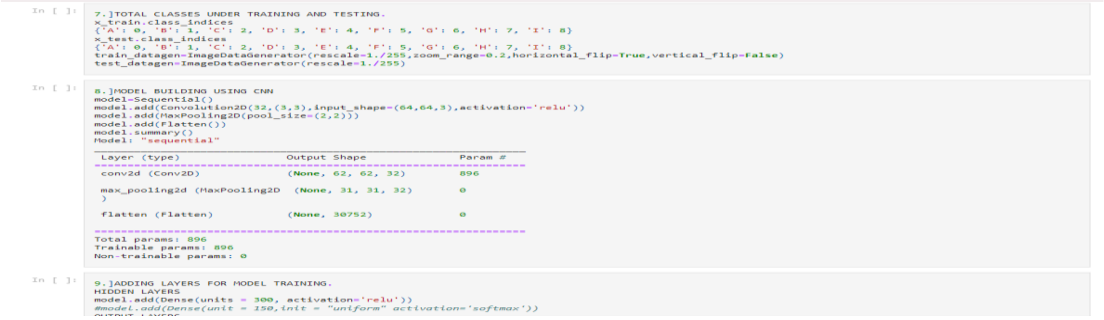
```
5.]UNZIPPING THE DATASET
from io import BytesIO
import zipfile
unzip=zipfile.ZipFile(BytesIO(streaming_body_6.read()),'r')
file_paths=unzip.namelist()
for path in file_paths:
    unzip.extract(path)
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
/tmp/wsuser/ipykernel_2521/251544276.py in
      1 from io import BytesIO
      2 import zipfile
----> 3 unzip=zipfile.ZipFile(BytesIO(streaming_body_6.read()),'r')
      4 file_paths=unzip.namelist()
      5 for path in file_paths:

NameError: name 'streaming_body_6' is not defined
ls
Dataset/
pwd
'/home/wsuser/work'
#checking that the dataset is there are not
import os
filenamer = os.listdir('/home/wsuser/work/Dataset/training_set')
```

In [ ]:

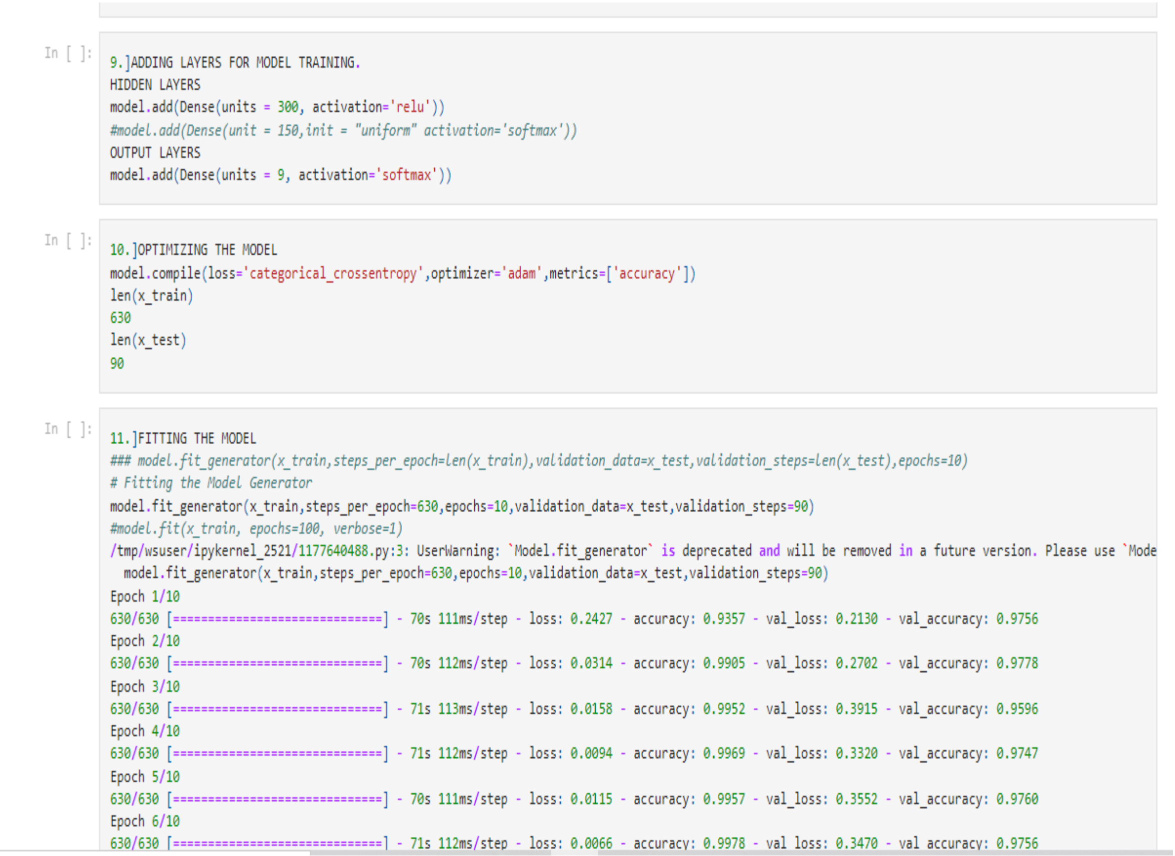
```
6.]TRAINING AND TESTING IMAGES UNDER CLASSES
x_train=train_datagen.flow_from_directory("/home/wsuser/work/Dataset/training_set",target_size=(64,64),class_mode="categorical",batch_size=25)
Found 15750 images belonging to 9 classes.
x_test=test_datagen.flow_from_directory("/home/wsuser/work/Dataset/test_set",target_size=(64,64),
class_mode='categorical' , batch_size=25)
Found 2250 images belonging to 9 classes.
```

In [ ]:

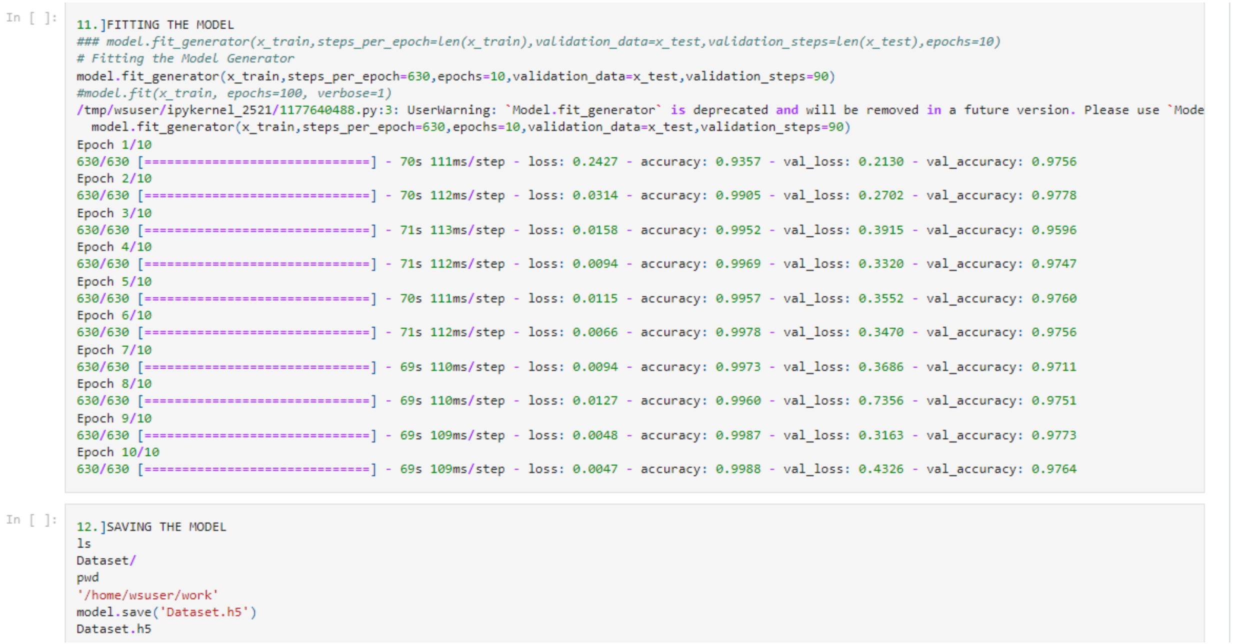
```
7.]TOTAL CLASSES UNDER TRAINING AND TESTING.
x_train.class_indices
```



```
In [ ]:
7.]TOTAL CLASSES UNDER TRAINING AND TESTING.
x_train.class_indices
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
x_test.class_indices
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
test_datagen=ImageDataGenerator(rescale=1./255)

In [ ]:
8.]MODEL BUILDING USING CNN
model=Sequential()
model.add(Convolution2D(32,(3,3),input_shape=(64,64,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.summary()
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 62, 62, 32)        896

 max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
 )

 flatten (Flatten)           (None, 30752)             0

=================================================================
Total params: 896
Trainable params: 896
Non-trainable params: 0

In [ ]:
9.]ADDING LAYERS FOR MODEL TRAINING.
HIDDEN LAYERS
model.add(Dense(units = 300, activation='relu'))
#model.add(Dense(unit = 150,init = "uniform" activation='softmax'))
OUTPUT LAYERS
```

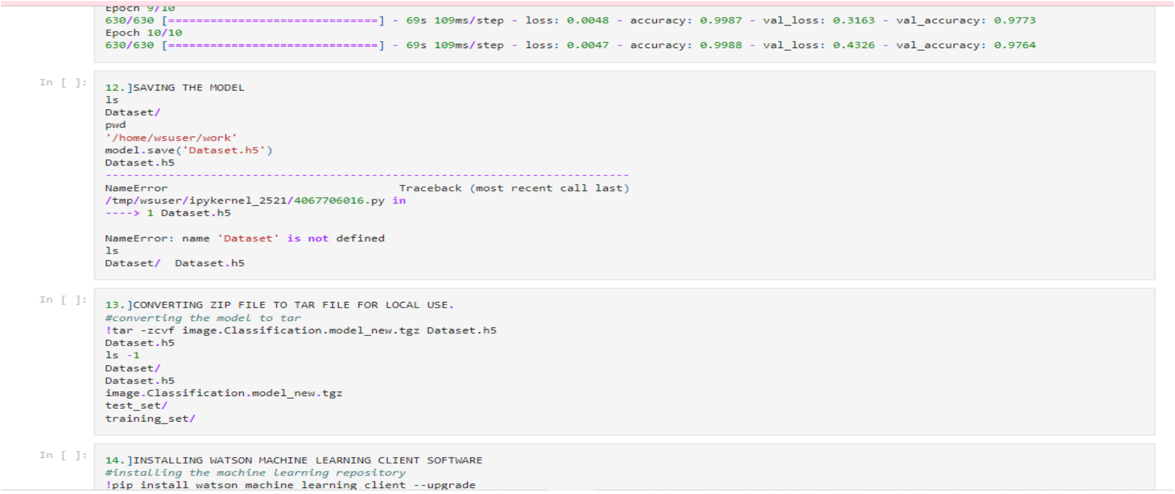


```
In [ ]:
9.]ADDING LAYERS FOR MODEL TRAINING.
HIDDEN LAYERS
model.add(Dense(units = 300, activation='relu'))
#model.add(Dense(unit = 150,init = "uniform" activation='softmax'))
OUTPUT LAYERS
model.add(Dense(units = 9, activation='softmax'))

In [ ]:
10.]OPTIMIZING THE MODEL
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
len(x_train)
630
len(x_test)
90

In [ ]:
11.]FITTING THE MODEL
### model.fit_generator(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test),epochs=10)
# Fitting the Model Generator
model.fit_generator(x_train,steps_per_epoch=630,epochs=10,validation_data=x_test,validation_steps=90)
#model.fit(x_train, epochs=100, verbose=1)
/tmp/wsuser/ipykernel_2521/1177640488.py:3: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Mode
  model.fit_generator(x_train,steps_per_epoch=630,epochs=10,validation_data=x_test,validation_steps=90)
Epoch 1/10
630/630 [==============================] - 70s 111ms/step - loss: 0.2427 - accuracy: 0.9357 - val_loss: 0.2130 - val_accuracy: 0.9756
Epoch 2/10
630/630 [==============================] - 70s 112ms/step - loss: 0.0314 - accuracy: 0.9905 - val_loss: 0.2702 - val_accuracy: 0.9778
Epoch 3/10
630/630 [==============================] - 71s 113ms/step - loss: 0.0158 - accuracy: 0.9952 - val_loss: 0.3915 - val_accuracy: 0.9596
Epoch 4/10
630/630 [==============================] - 71s 112ms/step - loss: 0.0094 - accuracy: 0.9969 - val_loss: 0.3320 - val_accuracy: 0.9747
Epoch 5/10
630/630 [==============================] - 70s 111ms/step - loss: 0.0115 - accuracy: 0.9957 - val_loss: 0.3552 - val_accuracy: 0.9760
Epoch 6/10
630/630 [==============================] - 71s 112ms/step - loss: 0.0066 - accuracy: 0.9978 - val_loss: 0.3470 - val_accuracy: 0.9756
```



```
In [ ]:
11.]FITTING THE MODEL
### model.fit_generator(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test),epochs=10)
# Fitting the Model Generator
model.fit_generator(x_train,steps_per_epoch=630,epochs=10,validation_data=x_test,validation_steps=90)
#model.fit(x_train, epochs=100, verbose=1)
/tmp/wsuser/ipykernel_2521/1177640488.py:3: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Mode
  model.fit_generator(x_train,steps_per_epoch=630,epochs=10,validation_data=x_test,validation_steps=90)
Epoch 1/10
630/630 [==============================] - 70s 111ms/step - loss: 0.2427 - accuracy: 0.9357 - val_loss: 0.2130 - val_accuracy: 0.9756
Epoch 2/10
630/630 [==============================] - 70s 112ms/step - loss: 0.0314 - accuracy: 0.9905 - val_loss: 0.2702 - val_accuracy: 0.9778
Epoch 3/10
630/630 [==============================] - 71s 113ms/step - loss: 0.0158 - accuracy: 0.9952 - val_loss: 0.3915 - val_accuracy: 0.9596
Epoch 4/10
630/630 [==============================] - 71s 112ms/step - loss: 0.0094 - accuracy: 0.9969 - val_loss: 0.3320 - val_accuracy: 0.9747
Epoch 5/10
630/630 [==============================] - 70s 111ms/step - loss: 0.0115 - accuracy: 0.9957 - val_loss: 0.3552 - val_accuracy: 0.9760
Epoch 6/10
630/630 [==============================] - 71s 112ms/step - loss: 0.0066 - accuracy: 0.9978 - val_loss: 0.3470 - val_accuracy: 0.9756
Epoch 7/10
630/630 [==============================] - 69s 110ms/step - loss: 0.0094 - accuracy: 0.9973 - val_loss: 0.3686 - val_accuracy: 0.9711
Epoch 8/10
630/630 [==============================] - 69s 110ms/step - loss: 0.0127 - accuracy: 0.9960 - val_loss: 0.7356 - val_accuracy: 0.9751
Epoch 9/10
630/630 [==============================] - 69s 109ms/step - loss: 0.0048 - accuracy: 0.9987 - val_loss: 0.3163 - val_accuracy: 0.9773
Epoch 10/10
630/630 [==============================] - 69s 109ms/step - loss: 0.0047 - accuracy: 0.9988 - val_loss: 0.4326 - val_accuracy: 0.9764
```

```
In [ ]:
12.]SAVING THE MODEL
ls
Dataset/
pwd
'/home/wsuser/work'
model.save('Dataset.h5')
Dataset.h5
```



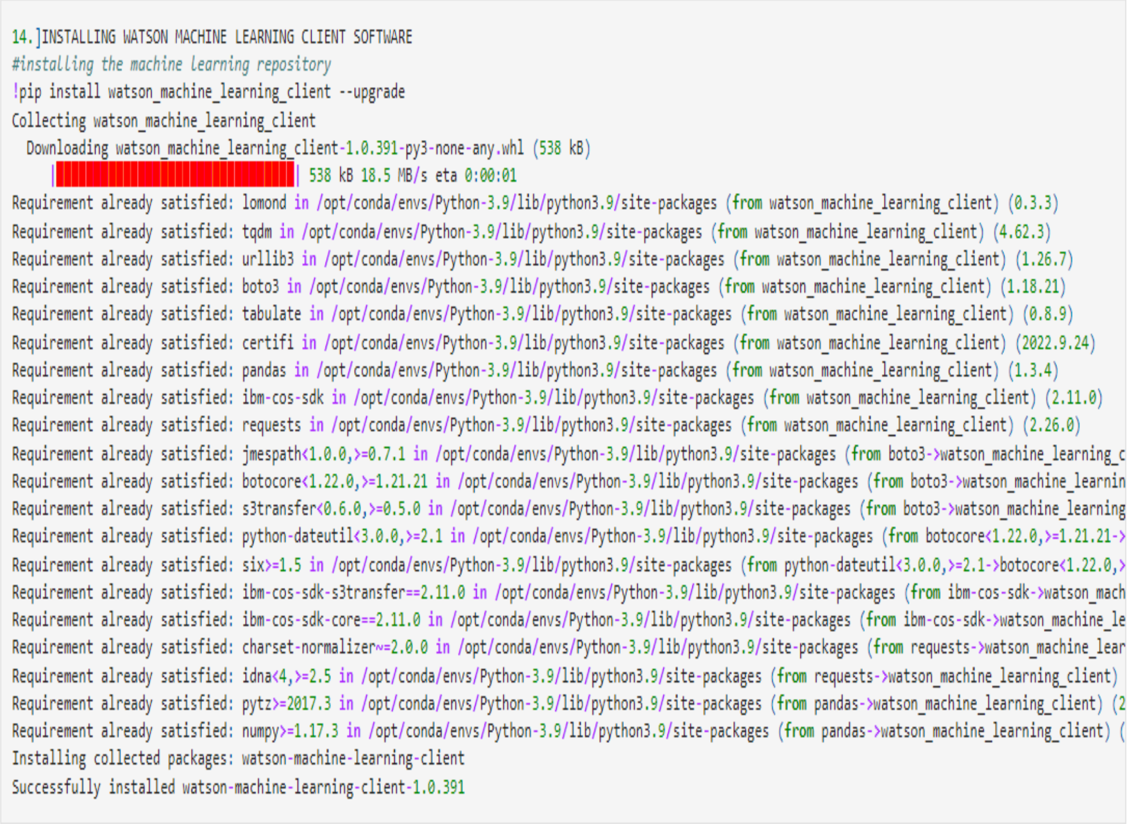
```
Epoch 9/10
630/630 [==============================] - 69s 109ms/step - loss: 0.0048 - accuracy: 0.9987 - val_loss: 0.3163 - val_accuracy: 0.9773
Epoch 10/10
630/630 [==============================] - 69s 109ms/step - loss: 0.0047 - accuracy: 0.9988 - val_loss: 0.4326 - val_accuracy: 0.9764
```

```
In [ ]:
12.]SAVING THE MODEL
ls
Dataset/
pwd
'/home/wsuser/work'
model.save('Dataset.h5')
Dataset.h5
```

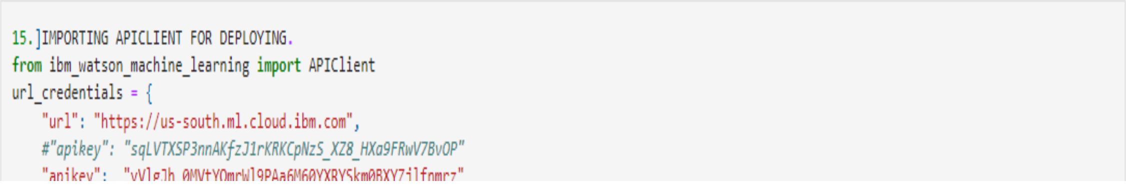
```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
/tmp/wsuser/ipykernel_2521/4067706016.py in
----> 1 Dataset.h5

NameError: name 'Dataset' is not defined
ls
Dataset/  Dataset.h5
```

```
In [ ]:
13.]CONVERTING ZIP FILE TO TAR FILE FOR LOCAL USE.
#converting the model to tar
!tar -zcvf image.Classification.model_new.tgz Dataset.h5
Dataset.h5
ls -1
Dataset/
Dataset.h5
image.Classification.model_new.tgz
test_set/
training_set/
```

```
In [ ]:
14.]INSTALLING WATSON MACHINE LEARNING CLIENT SOFTWARE
#installing the machine learning repository
!pip install watson machine learning client --upgrade
```



```
14.]INSTALLING WATSON MACHINE LEARNING CLIENT SOFTWARE
#installing the machine learning repository
!pip install watson_machine_learning_client --upgrade
Collecting watson_machine_learning_client
  Downloading watson_machine_learning_client-1.0.391-py3-none-any.whl (538 kB)
     |████████████████████████████████| 538 kB 18.5 MB/s eta 0:00:01
Requirement already satisfied: lomond in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson_machine_learning_client) (0.3.3)
Requirement already satisfied: tqdm in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson_machine_learning_client) (4.62.3)
Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson_machine_learning_client) (1.26.7)
Requirement already satisfied: boto3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson_machine_learning_client) (1.18.21)
Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson_machine_learning_client) (0.8.9)
Requirement already satisfied: certifi in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson_machine_learning_client) (2022.9.24)
Requirement already satisfied: pandas in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson_machine_learning_client) (1.3.4)
Requirement already satisfied: ibm-cos-sdk in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson_machine_learning_client) (2.11.0)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson_machine_learning_client) (2.26.0)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson_machine_learning_c
Requirement already satisfied: botocore<1.22.0,>=1.21.21 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson_machine_learnin
Requirement already satisfied: s3transfer<0.6.0,>=0.5.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson_machine_learning
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from botocore<1.22.0,>=1.21.21->
Requirement already satisfied: six>=1.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1->botocore<1.22.0,>
Requirement already satisfied: ibm-cos-sdk-s3transfer==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk->watson_mach
Requirement already satisfied: ibm-cos-sdk-core==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk->watson_machine_le
Requirement already satisfied: charset-normalizer~=2.0.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->watson_machine_lear
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->watson_machine_learning_client)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas->watson_machine_learning_client) (2
Requirement already satisfied: numpy>=1.17.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas->watson_machine_learning_client) (
Installing collected packages: watson-machine-learning-client
Successfully installed watson-machine-learning-client-1.0.391
```



```
15.]IMPORTING APICLIENT FOR DEPLOYING.
from ibm_watson_machine_learning import APIClient
url_credentials = {
    "url": "https://us-south.ml.cloud.ibm.com",
    #"apikey": "sqLVTXSP3nnAKfzJ1rKRKCpNzS_XZ8_HXa9FRwV7BvOP"
    "apikey":  "yVlgJh_0MVtYOmrWl9PAa6M60YXRYSkm0BXYZilfomrz"
```

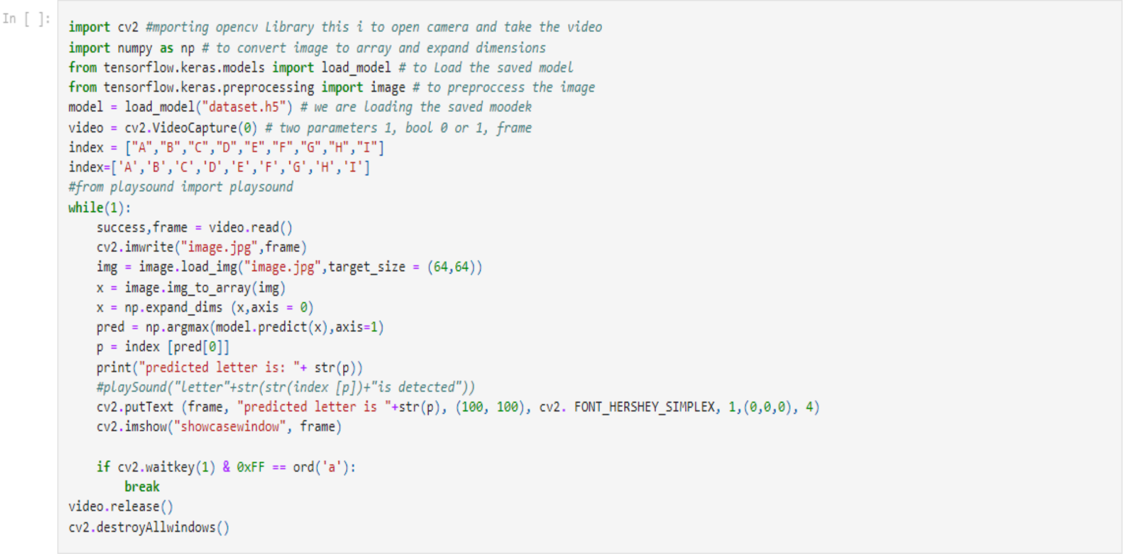
In [ ]:

```
15.]IMPORTING APICLIENT FOR DEPLOYING.
from ibm_watson_machine_learning import APIClient
url_credentials = {
    "url": "https://us-south.ml.cloud.ibm.com",
    #"apikey": "sqLVTXSP3nnAKfzJ1rKRKCpNzS_XZ8_HXa9FRwV7BvOP"
    "apikey":  "yVlgJh_0MVtYQmrWl9PAa6M60YXRYSkm0BXYZjlfnmrz"
}
client = APIClient(url_credentials)
client = APIClient(url_credentials)
client
```

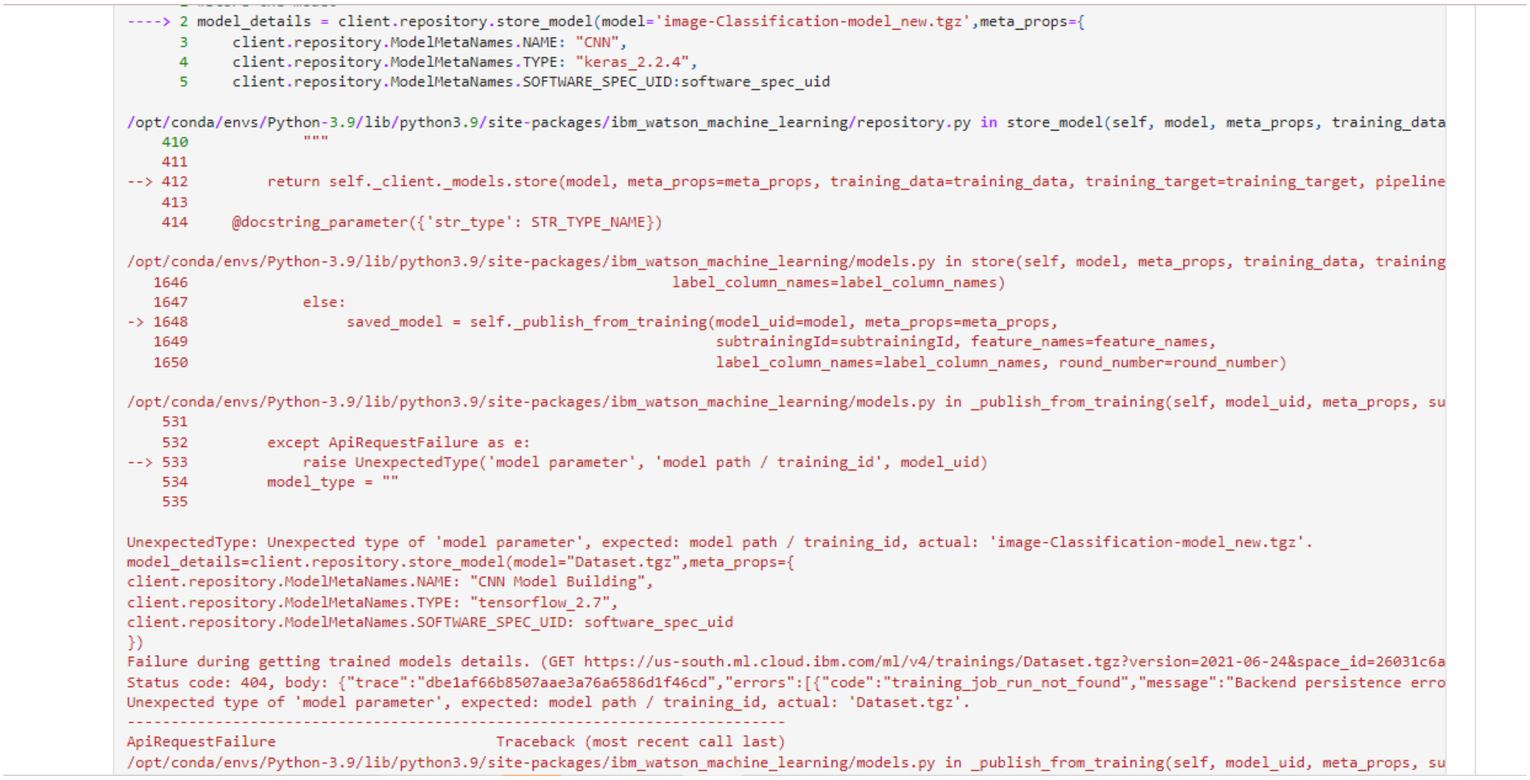
In [ ]:

```
16.]CREATING API_CLIENT SPACE ID.
def guid_from_space_name(client, space_name):
    space = client.spaces.get_details()
    return(next(item for item in space['resources'] if item['entity']['name'] == space_name)['metadata']['id'])
space_uid = guid_from_space_name(client, 'newspace')
print("space UID = " + space_uid)
space UID = 26031c6a-3567-437f-9ccb-d8ca0f32a42f
client.set.default_space(space_uid)
'SUCCESS'
client.software_specifications.list(500)
------------------------------  ------------------------------------  ----
NAME                            ASSET_ID                              TYPE
default_py3.6                   0062b8c9-8b7d-44a0-a9b9-46c416adcbd9  base
kernel-spark3.2-scala2.12       020d69ce-7ac1-5e68-ac1a-31189867356a  base
pytorch-onnx_1.3-py3.7-edt      069ea134-3346-5748-b513-49120e15d288  base
scikit-learn_0.20-py3.6         09c5a1d0-9c1e-4473-a344-eb7b665ff687  base
spark-mllib_3.0-scala_2.12      09f4cff0-90a7-5899-b9ed-1ef348aebdee  base
pytorch-onnx_rt22.1-py3.9       0b848dd4-e681-5599-be41-b5f6fccc6471  base
ai-function_0.1-py3.6           0cdb0f1e-5376-4f4d-92dd-da3b69aa9bda  base
shiny-r3.6                      0e6e79df-875e-4f24-8ae9-62dcc2148306  base
tensorflow_2.4-py3.7-horovod    1092590a-307d-563d-9b62-4eb7d64b3f22  base
pytorch_1.1-py3.6               10ac12d6-6b30-4ccd-8392-3e922c096a92  base
tensorflow_1.15-py3.6-ddl       111e41b3-de2d-5422-a4d6-bf776828c4b7  base
autoai-kb_rt22.2-py3.10         125b6d9a-5b1f-5e8d-972a-b251688ccf40  base
```

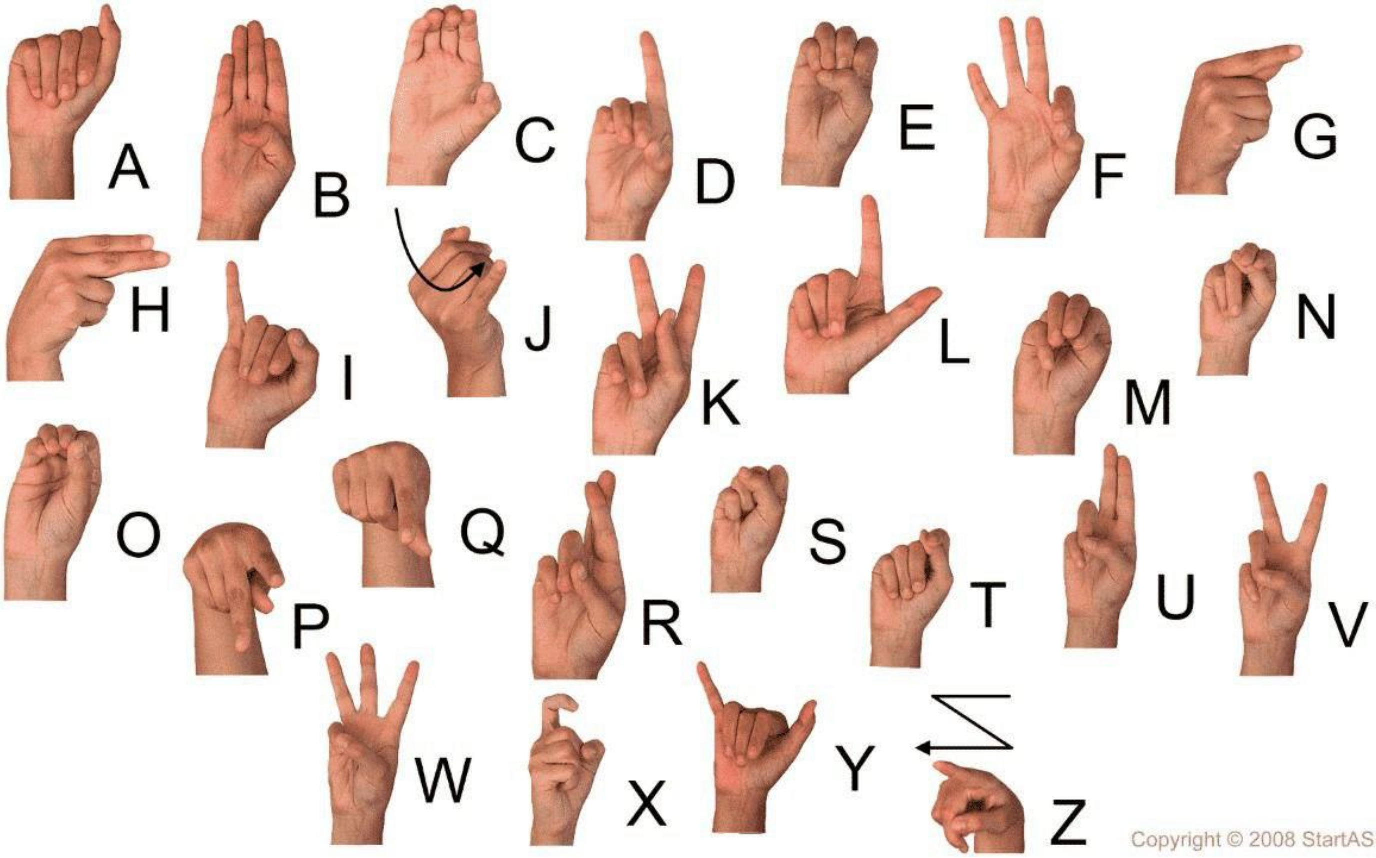
```
autoai-ts_3.8-py3.8              2aa0c932-798f-5ae9-abd6-15e0c2402fb5  base
tensorflow_1.15-py3.6            2b73a275-7cbf-420b-a912-eae7f436e0bc  base
kernel-spark3.3-py3.9            2b7961e2-e3b1-5a8c-a491-482c8368839a  base
pytorch_1.2-py3.6                2c8ef57d-2687-4b7d-acce-01f94976dac1  base
spark-mllib_2.3                  2e51f700-bca0-4b0d-88dc-5c6791338875  base
pytorch-onnx_1.1-py3.6-edt       32983cea-3f32-4400-8965-dde874a8d67e  base
spark-mllib_3.0-py37             36507ebe-8770-55ba-ab2a-eafe787600e9  base
spark-mllib_2.4                  390d21f8-e58b-4fac-9c55-d7ceda621326  base
autoai-ts_rt22.2-py3.10          396b2e83-0953-5b86-9a55-7ce1628a406f  base
xgboost_0.82-py3.6               39e31acd-5f30-41dc-ae44-60233c80306e  base
pytorch-onnx_1.2-py3.6-edt       40589d0e-7019-4e28-8daa-fb03b6f4fe12  base
pytorch-onnx_rt22.2-py3.10       40e73f55-783a-5535-b3fa-0c8b94291431  base
default_r36py38                  41c247d3-45f8-5a71-b065-8580229facf0  base
autoai-ts_rt22.1-py3.9           4269d26e-07ba-5d40-8f66-2d495b0c71f7  base
autoai-obm_3.0                   42b92e18-d9ab-567f-988a-4240ba1ed5f7  base
pmml-3.0_4.3                     493bcb95-16f1-5bc5-bee8-81b8af80e9c7  base
spark-mllib_2.4-r_3.6            49403dff-92e9-4c87-a3d7-a42d0021c095  base
xgboost_0.90-py3.6               4ff8d6c2-1343-4c18-85e1-689c965304d3  base
pytorch-onnx_1.1-py3.6           50f95b2a-bc16-43bb-bc94-b0bed208c60b  base
autoai-ts_3.9-py3.8              52c57136-80fa-572e-8728-a5e7cbb42cde  base
spark-mllib_2.4-scala_2.11       55a70f99-7320-4be5-9fb9-9edb5a443af5  base
spark-mllib_3.0                  5c1b0ca2-4977-5c2e-9439-ffd44ea8ffe9  base
autoai-obm_2.0                   5c2e37fa-80b8-5e77-840f-d912469614ee  base
spss-modeler_18.1                5c3cad7e-507f-4b2a-a9a3-ab53a21dee8b  base
cuda-py3.8                       5d3232bf-c86b-5df4-a2cd-7bb870a1cd4e  base
autoai-kb_3.1-py3.7              632d4b22-10aa-5180-88f0-f52dfb6444d7  base
pytorch-onnx_1.7-py3.8           634d3cdc-b562-5bf9-a2d4-ea90a478456b  base
spark-mllib_2.3-r_3.6            6586b9e3-ccd6-4f92-900f-0f8cb2bd6f0c  base
tensorflow_2.4-py3.7             65e171d7-72d1-55d9-8ebb-f813d620c9bb  base
spss-modeler_18.2                687eddc9-028a-4117-b9dd-e57b36f1efa5  base
pytorch-onnx_1.2-py3.6           692a6a4d-2c4d-45ff-a1ed-b167ee55469a  base
spark-mllib_2.3-scala_2.11       7963efe5-bbec-417e-92cf-0574e21b4e8d  base
spark-mllib_2.4-py37             7abc992b-b685-532b-a122-a396a3cdbaab  base
caffe_1.0-py3.6                  7bb3dbe2-da6e-4145-918d-b6d84aa93b6b  base
pytorch-onnx_1.7-py3.7           812c6631-42b7-5613-982b-02098e6c909c  base
cuda-py3.6                       82c79ece-4d12-40e6-8787-a7b9e0f62770  base
tensorflow_1.15-py3.6-horovod    8964680e-d5e4-5bb8-919b-8342c6c0dfd8  base
hybrid_0.1                       8c1a58c6-62b5-4dc4-987a-df751c2756b6  base
```



```
In [ ]:
import cv2 #mporting opencv Library this i to open camera and take the video
import numpy as np # to convert image to array and expand dimensions
from tensorflow.keras.models import load_model # to Load the saved model
from tensorflow.keras.preprocessing import image # to preproccess the image
model = load_model("dataset.h5") # we are loading the saved moodek
video = cv2.VideoCapture(0) # two parameters 1, bool 0 or 1, frame
index = ["A","B","C","D","E","F","G","H","I"]
index=['A','B','C','D','E','F','G','H','I']
#from playsound import playsound
while(1):
    success,frame = video.read()
    cv2.imwrite("image.jpg",frame)
    img = image.load_img("image.jpg",target_size = (64,64))
    x = image.img_to_array(img)
    x = np.expand_dims (x,axis = 0)
    pred = np.argmax(model.predict(x),axis=1)
    p = index [pred[0]]
    print("predicted letter is: "+ str(p))
    #playSound("Letter"+str(str(index [p])+"is detected"))
    cv2.putText (frame, "predicted letter is "+str(p), (100, 100), cv2. FONT_HERSHEY_SIMPLEX, 1,(0,0,0), 4)
    cv2.imshow("showcasewindow", frame)

    if cv2.waitkey(1) & 0xFF == ord('a'):
        break
video.release()
cv2.destroyAllwindows()
```



```
----> 2 model_details = client.repository.store_model(model='image-Classification-model_new.tgz',meta_props={
      3     client.repository.ModelMetaNames.NAME: "CNN",
      4     client.repository.ModelMetaNames.TYPE: "keras_2.2.4",
      5     client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ibm_watson_machine_learning/repository.py in store_model(self, model, meta_props, training_data
    410             """
    411 
--> 412         return self._client._models.store(model, meta_props=meta_props, training_data=training_data, training_target=training_target, pipeline
    413 
    414     @docstring_parameter({'str_type': STR_TYPE_NAME})

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ibm_watson_machine_learning/models.py in store(self, model, meta_props, training_data, training
   1646                                                        label_column_names=label_column_names)
   1647             else:
-> 1648                 saved_model = self._publish_from_training(model_uid=model, meta_props=meta_props,
   1649                                                             subtrainingId=subtrainingId, feature_names=feature_names,
   1650                                                             label_column_names=label_column_names, round_number=round_number)

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ibm_watson_machine_learning/models.py in _publish_from_training(self, model_uid, meta_props, su
    531 
    532         except ApiRequestFailure as e:
--> 533             raise UnexpectedType('model parameter', 'model path / training_id', model_uid)
    534         model_type = ""
    535 

UnexpectedType: Unexpected type of 'model parameter', expected: model path / training_id, actual: 'image-Classification-model_new.tgz'.
model_details=client.repository.store_model(model="Dataset.tgz",meta_props={
client.repository.ModelMetaNames.NAME: "CNN Model Building",
client.repository.ModelMetaNames.TYPE: "tensorflow_2.7",
client.repository.ModelMetaNames.SOFTWARE_SPEC_UID: software_spec_uid
})
Failure during getting trained models details. (GET https://us-south.ml.cloud.ibm.com/ml/v4/trainings/Dataset.tgz?version=2021-06-24&space_id=26031c6a
Status code: 404, body: {"trace":"dbe1af66b8507aae3a76a6586d1f46cd","errors":[{"code":"training_job_run_not_found","message":"Backend persistence erro
Unexpected type of 'model parameter', expected: model path / training_id, actual: 'Dataset.tgz'.
---------------------------------------------------------------------------
ApiRequestFailure                         Traceback (most recent call last)
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ibm_watson_machine_learning/models.py in _publish_from_training(self, model_uid, meta_props, su
```


A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

GITHUB

https://github.com/IBM-EPBL/IBM-Project-52118-1660989503