NATURAL DISASTER INTENSITY ANALYSIS AND CLASSIFICATION USING ARTIFICIAL INTELLIGENCE

PROJECT REPORT

Submitted by

Team ID:PNT2022TMID38277

D.GAYATHRI(412619104011)

S.DIVYA(412619104010)

P.SUJASHNI(412619104044)

A.KOKILA(412619104018)

Under the guidance of ,

Mr.Anburaman.S(Mentor)

Ms.Swathi(Industry mentor)

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

Natural disasters not only disturb the human ecological system but also destroy the properties and critical infrastructures of human societies and even lead to permanent change in the ecosystem. Disaster can be caused by naturally occurring events such as earthquakes, cyclones, floods, and wildfires. Many deep learning techniques have been applied by various researchers to detect and classify natural disasters to overcome losses in ecosystems, but detection of natural disasters still facesissues due to the complex and imbalanced structures of images. To tackle this problem,we developed a multilayered deep convolutional neural network model that classifies the natural disaster and tells the intensity of disaster of natural The model uses an integrated webcam to capture the video frame and the video frame is compared with thePre-trained model and the type of disaster is identified and showcased on the OpenCV window.

1.2 PURPOSE

Disaster management plays an integral role in keeping communities safe. It involves coordinating the resources, such as pollution control systems, and responsibilities, such as following best practice policies, needed to prevent, prepare for,respond to, and recover from emergencies.Natural disasters generally constitute an emergency since they require immediate

intervention due to their high impact on human health and safety; they affect the normal functioning of working infrastructure, interrupting normal day activities and representing a risk for residents and workers in affected areas.

CHAPTER2

LITERATURE SURVEY

2.1 EXISTING PROBLEM

Many researchers have attempted to use different deep learning methods for detection of natural disasters. However, the detection of natural disasters by using deeplearning techniques still faces various issues due to noise and serious class imbalance problems. To address these problems, we proposed a multilayered deep convolutional neural network for detection and intensity classification of natural disasters. The proposed method works in two blocks—one for detection of natural disaster occurrenceand the second block is used to remove imbalanced class issues. The results were calculated as average statistical values: sensitivity, 97.54%; specificity, 98.22%; accuracy rate, 99.92%; precision, 97.79%; and F1- score, 97.97% for the proposed model. The proposed model achieved the highest accuracy as compared to other state-of-the-art methods due to its multilayered structure. The proposed model performs significantly better for natural disaster detection and classification, but in the future the model can be used for various natural disaster detection processes.

2.2 REFERENCES

1. Mignan, A.; Broccardo, M. Neural network applications in earthquake prediction (1994–2019): Meta-analytic and statistical insights on their limitations. Seism.Res. Lett. 2020, 91, 2330–2342. [CrossRef]

2. Tonini, M.; D’Andrea, M.; Biondi, G.; Degli Esposti, S.; Trucchia, A.; Fiorucci, P.A Machine Learning-Based Approach forWildfire Susceptibility Mapping.

2.3 PROBLEM STATEMENT DEFINITION

"IMD (Indian Meteorology department) is responsible to issue warnings for the rainfall and CWC (Central Water Commission) keeps a record of water reservoirs, however there is a lack of collation of data issued from both these departments. This prevents us from determining the impact/seriousness and due to which there are times where adequate forewarnings are not provided. There are several High rainfall areas, low lying areas or flood prone areas. Currently there are limitations that these areas cannot be alerted before the critical situation because of the data unavailability or unavailability of simulation models which can calculate and predict the data. There is a requirement of data on the area likely to be inundated(depth) by release of water from reservoirs. 3D models may help in calculation of such data.a) Adequate forewarning forthe area where floods are likely to occur. b) Low lying areas may be alerted about the release of accurate quantity of water from the reservoirs and thus evacuation/shifting ofthe people can be planned. c) It will help the Response forces to deploy their resources accordingly d) Prediction of release of water based on rainfall in catchment area and dissemination of information to the affected public through mobile and other mediums.

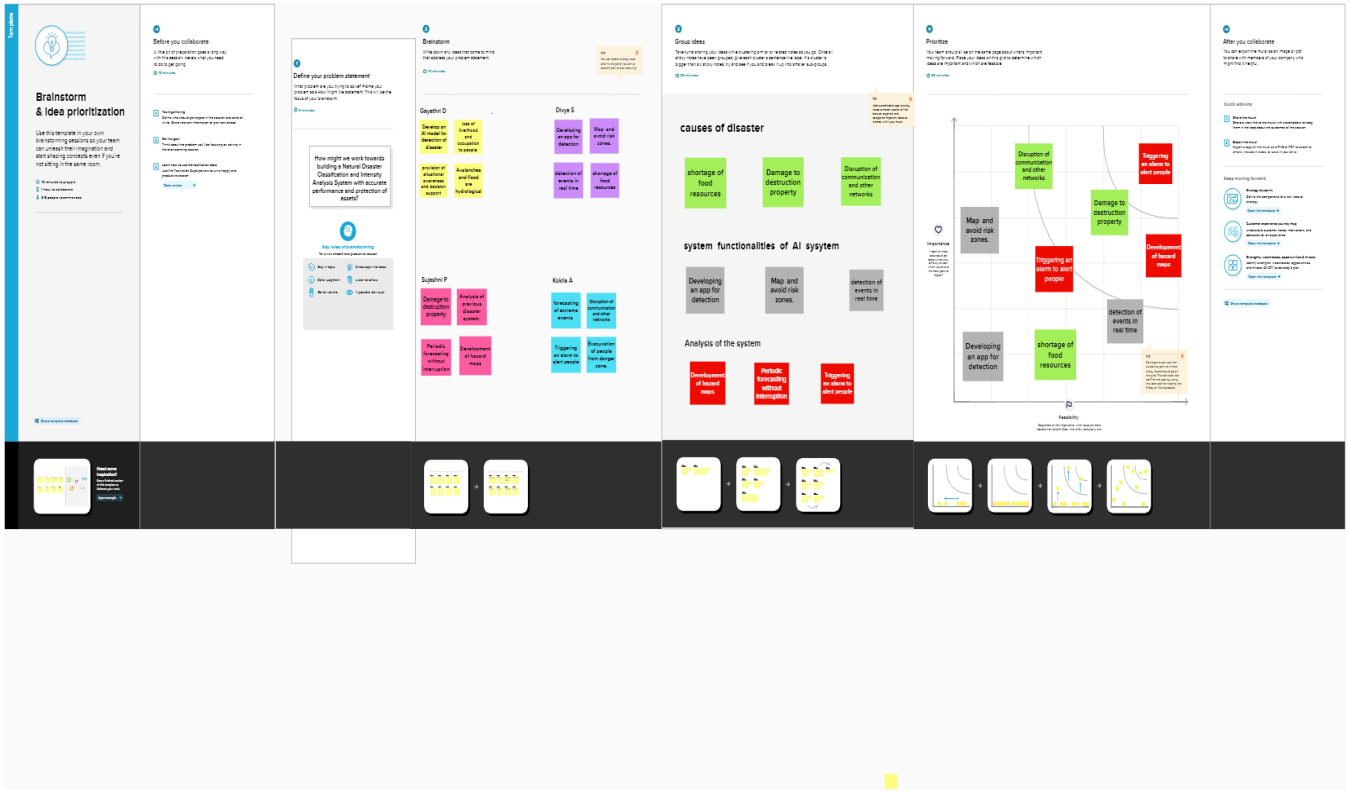
CHAPTER 3

IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

NATURAL DISASTERS INTENSITY ANALYSIS AND CLASSIFICATION USING AI

3.2 IDEATION & BRAINSTORMING



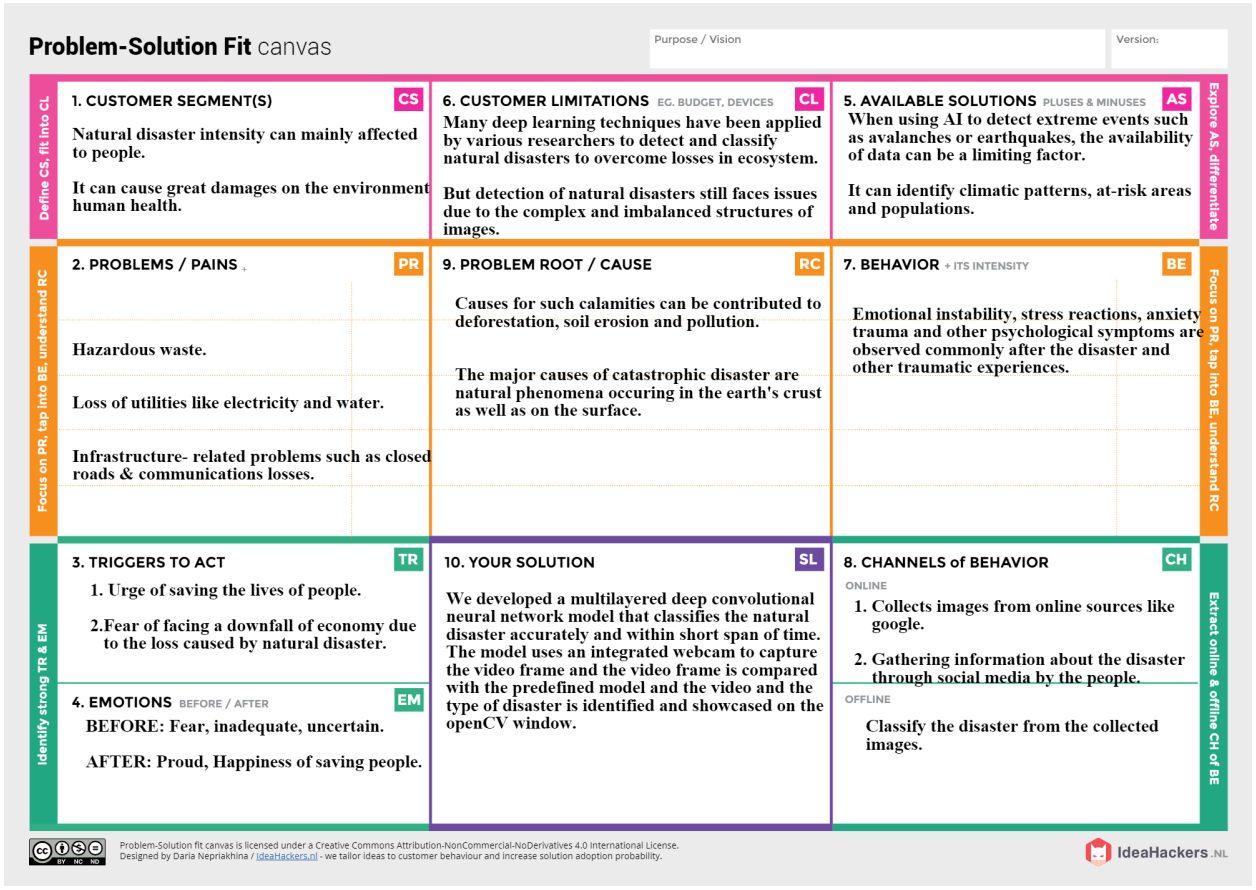
3.3 PROPOSED SOLUTION

Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To monitoring and predicting the disasters and its intensity of impacts on the region.
2.	Idea / Solution description	To use classification algorithm to identify the impacts of disaster.
3.	Novelty / Uniqueness	A Natural disaster is" the negative impact following an actual occurrence of natural hazard in the event that is significantly harms a community".
4.	Social Impact / Customer Satisfaction	Copying capacity,culturals impacts,loss of livelihood,loss absorption,loss acceptance,social vulnerability.
5.	Business Model (Revenue Model)	Revenue generated through Royalty payments, product license costs in department , research and educational platforms.
6.	Scalability of the Solution	A first scalable implicit solver for nonlinear time-evolution earthquakes city problem on low ordered unstructured finite elements with artificial intelligence.

3.4 PROBLEM SOLUTION FIT



Problem-Solution Fit canvas

Purpose / Vision

Version:

Define CS, fit into CL

1. CUSTOMER SEGMENT(S) CS

Natural disaster intensity can mainly affected to people.

It can cause great damages on the environment human health.

6. CUSTOMER LIMITATIONS EG. BUDGET, DEVICES CL

Many deep learning techniques have been applied by various researchers to detect and classify natural disasters to overcome losses in ecosystem.

But detection of natural disasters still faces issues due to the complex and imbalanced structures of images.

5. AVAILABLE SOLUTIONS PLUSES & MINUSES AS

When using AI to detect extreme events such as avalanches or earthquakes, the availability of data can be a limiting factor.

It can identify climatic patterns, at-risk areas and populations.

Explore AS, differentiate

Focus on PR, tap into BE, understand RC

2. PROBLEMS / PAINS PR

Hazardous waste.

Loss of utilities like electricity and water.

Infrastructure- related problems such as closed roads & communications losses.

9. PROBLEM ROOT / CAUSE RC

Causes for such calamities can be contributed to deforestation, soil erosion and pollution.

The major causes of catastrophic disaster are natural phenomena occuring in the earth's crust as well as on the surface.

7. BEHAVIOR + ITS INTENSITY BE

Emotional instability, stress reactions, anxiety trauma and other psychological symptoms are observed commonly after the disaster and other traumatic experiences.

Focus on PR, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TO ACT TR

1. Urge of saving the lives of people.
2. Fear of facing a downfall of economy due to the loss caused by natural disaster.

4. EMOTIONS BEFORE / AFTER EM

BEFORE: Fear, inadequate, uncertain.

AFTER: Proud, Happiness of saving people.

10. YOUR SOLUTION SL

We developed a multilayered deep convolutional neural network model that classifies the natural disaster accurately and within short span of time. The model uses an integrated webcam to capture the video frame and the video frame is compared with the predefined model and the video and the type of disaster is identified and showcased on the openCV window.

8. CHANNELS of BEHAVIOR CH

ONLINE

1. Collects images from online sources like google.
2. Gathering information about the disaster through social media by the people.

OFFLINE

Classify the disaster from the collected images.

Extract online & offline CH of BE

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Sub Requirement (Story / Sub-Task)
FR-1	**Tsunami :** A series of large waves of extremely long wavelength and period usually generated by a violent, impulsive undersea disturbance or activity near the coast or in the ocean. When a sudden displacement of a large volume of water occurs, or if the seafloor is suddenly raised or dropped by an earthquake, big tsunami waves can be formed.
FR-2	**Earthquake :** Any sudden shaking of the ground caused by the passage of seismic waves through Earth's rocks. Seismic waves are produced when some form of energy stored in Earth's crust is suddenly released, usually when masses of rock straining against one another suddenly fracture and "slip."
FR-3	**Droughts:** The primary cause of any drought is efficiency of rainfall and in particular , the timing, distribution and intensity of this deficiency in relation to existing reserves.
FR-4	**Tropical cyclones:** The major natural disaster that affects the coastal regions of India is cyclone and has a coastline of about 7516 kilometres , it is exposed to nearly 10% of worlds tropical cyclones.
FR-5	**Landslides:** It mainly affects the Himalayan region and the western ghats of India. Landslides are also common in the nilgiri range. It is estimated that 30 percent of the world's landslides occur in the Himalayas. The Himalayan mountains which constitute the youngest and most dominating mountain system in the world.

FR-6	**Volcanoes:** A volcano is an opening in the earth's crust through which lava, volcanic ash, and gases escape. Volcanic eruptions are partly driven by pressure from dissolved gas, much as escaping gases force the cork out of a bottle of champagne.

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

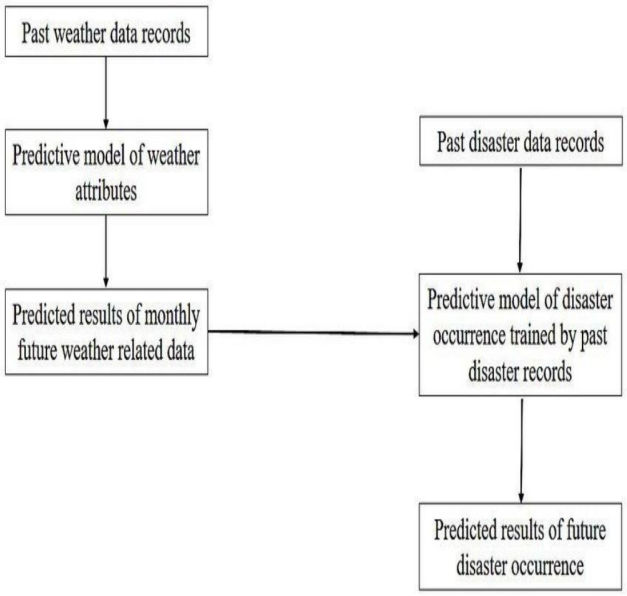
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The wide spectrum of technologies used in Geographical Information System, Global Positioning System (GPS), Satellite navigation system, Satellite communication.
NFR-2	**Security**	Identification and measuring disaster risk. Incorporating DRM into national planning and investment.
NFR-3	**Reliability**	Disaster-related damages are typically measured by separately examining the numbers of fatalities, injuries.
NFR-4	**Performance**	The identification of hazards; a review of the technical characteristics of hazards such as their location, intensity, frequency and probability.
NFR-5	**Availability**	The number and cost of weather and climate disasters is rising due to a combination of population growth and development along with the influence of human-caused climate change.

CHAPTER 5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAM

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2 USER STORIES

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

6.1 6.2 SPRINT DELIVERY SCHEDULE

Sprint	Total Story Points	Duration	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Releas e Date (Actua l)
Sprint -1	14	6 days	24 Oct 2022	29 Oct 2022	30 Oct 2022
Sprint- 2	12	6 days	31 Oct 2022	5 Nov2022	6 Nov 2022
Sprint - 3	6	6 days	07 Nov2022	12 Nov2022	8 Nov 2022
Sprint - 4	6	6 days	14 Nov2022	19 Nov2022	20 Nov 2022

7.HTML CODE

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width,
initial-scale=1.0">
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.
min.
css"
rel="stylesheet" integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5lDxbcnCeuOxjzrPF/et3URy9
Bv1WTRi">
<title>Document</title>
</head>
<body>
<div class="card text-center">
<div class="card-header">
```

```
<ul class="nav nav-tabs card-header-tabs">
<li class="nav-item">
<a class="nav-link active" aria-current="true" href="home.html"
style="font-size:
24px;">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" href="intro.html" style="font-size:
24px;">Introduction</a>
</li>
<li class="nav-item">
<a class="nav-link" href="upload.html" style="font-size:
24px;">Upload</a>
</li>
</ul>
<h3 style="float: right;">AI based Natural Disaster Analysis</h3>
</div>
</div>
<div class = "container" style="text-align: center;">
<div class="card" style="width: 18rem; padding: 10px; margin: 40px;
margin-left:
40px;display:inline-block">
```

```
<img class="card-img-top" src="{{ url_for('static',
filename='cyclone.jpg') }}" alt="Cardimage cap">
<div class="card-body" >
<h5 class="card-title">Cyclone</h5>
<p class="card-text">cyclone, large system of winds that circulates
counterclockwise directionnorth of the Equator and clockwise
direction to
the south.</p>
<a href="https://en.wikipedia.org/wiki/Cyclone" class="btn
btn-primary">Know more</a></div>
</div>
<div class="card" style="width: 18rem; padding: 10px; margin: 40px;
margin-left:
40px;display:inline-block">
<img class="card-img-top" src="{{ url_for('static',
filename='earthquake.jpg') }}" alt="Cardimage cap">
<div class="card-body" >
<h5 class="card-title">Earthquake</h5>
<p class="card-text">A sudden violent shaking of the ground,
causing great
destruction, as aresult of movements within the earth's crust.</p>
<a href="https://en.wikipedia.org/wiki/Earthquake" class="btn
```

```
btn-primary">Knowmore</a>
</div>
</div>
</div>
<div class = "container" style="text-align: center;">
<div class="card" style="width: 18rem; padding: 10px; margin: 40px;
margin-left:
40px;display:inline-block">
<img class="card-img-top" src="{{ url_for('static',
filename='flood.jpg')
}}" alt="Cardimagecap">
<div class="card-body" >
<h5 class="card-title">Flood</h5>
<p class="card-text">An overflow of a large amount of water beyond
its
normal limits, especially over what is normally dry land.</p>
<a href="https://en.wikipedia.org/wiki/Flood" class="btn
btn-primary">know
more</a></div>
</div>
<div class="card" style="width: 18rem; padding: 10px; margin: 40px;
margin-left:
```

```
40px;display:inline-block">
<img class="card-img-top" src="{{
url_for('static',filename='wildfire.jpg') }}" alt="Cardimage cap">
<div class="card-body" >
<h5 class="card-title">Wild Fire</h5>
<p class="card-text">A wildfire is an unplanned, uncontrolled and
unpredictable fire in area of combustible vegetation starting in rural
and urban areas.</p>
a href="https://en.wikipedia.org/wiki/Wildfire" class="btn
btn-primary">Know more</a></div>
</div>
</div>
</body>
</html
```

PYTHON CODE

```
# -*-
coding:
utf-8
-*-
                """Build Python Code & Run the Application


                Automatically generated by Colaboratory.


                Original file is located at


                https://colab.research.google.com/drive/1TxsPz40WJMI7I8puz5Q
                G3LOHXUQF36Us
                """



                from flask import Flask, render_template


                app = Flask(__name__)



                @app.route('/')

                def home():

                   return render_template('homepage.html', title='Disaster
                Classifier | Home', active_page='home')


                @app.route('/intro')

                def intro():

                   return render_template('intro.html', title='Disaster
                Classifier | About', active_page='intro')


                @app.route('/launch')

                def launch():
```

```
    return render_template('launch.html', title='Disaster
Classifier | Launch', active_page='launch')


if __name__ == '__main__':
    app.run(debug=True)
```

FUTURE SCOPE

AI data setups are trained to predict seismic data to analyze the patterns of earthquake occurrences, rainfall records and monitor flooding, measure the intensity of hurricanes and read the geological data to understand volcanic eruptions, such systemscan reduce the catastrophic impact of natural disasters.

Last year, Google's Pilot project to monitor flood in India with the help of AI, wasa successful one – it was a Patna project. They were able to predict floods and the regions that it would be affected due to the natural disaster with an accuracy of over 90%. It was possible owing to the combination of data from government agencies that provide on-ground information – from measuring devices placed on the spot and satellite captured images of flood-prone areas. They ran hundreds of thousands of simulations on its machine learning (ML) models to predict the flow of water.In the future, leveraging AI can help disaster management bodies install drones, sensors androbots to provide accurate information about damaged buildings and landscapes, potential floods, making rescue missions safer and less time-consuming. There is a need for smart technology to be integrated within our local communities. Immediate response and tech-based solutions can help reduce the extentof damage. However,

since AI is based on machine codes, there is a scope of limitations and errors. However, the amalgamation of human, empathy and alertness, could do wonders in the field of crisis management.

CHAPTER 8

PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

Sprint	Functional Requirement(Epic)	User story Number	User story / Task	Story points	Priority	Team members
Sprint-1	Registration	USN – 1	As a user, Registering into the product usinga valid email address	5	High	Gayathri D
Sprint-2	Registration	USN – 2	As a user, Registering into the product usinga valid username andpassword	3	Medium	Kokila A Sujashni P
Sprint-1	Authentication	USN – 3	As a user , I adept to logginginto the system with credentials	4	High	Divya S Gayathri D
Sprint-2	Authentication	USN - 4	As a user , I adept to logging intothe systemwithOTP	2	High	Divya S, Gayathri D Kokila A Sujashni P
Sprint-1	Designation ofRegion	USN – 5	selecting the region ofinterest to be monitored and analysed	3	High	Divya s Sujashni P

						Gayathri D
Sprint-2	Analysis of Required Phenomenon	USN – 6	Regulating certain factors influencing theactions of the phenomenon	3	High	Gayathri D Kokila A
Sprint-2	Accumulation ofrequired Data	USN – 7	Gathering data and detailed report onpastevent analysis	4	Medium	Kokila A Sujashni P

Sprint-4	Organizing Unstructureddata	USN – 8	Organizing and reorienting the raw data into a refined data	3	Low	Gayathri D Divya S
Sprint-2	AlgoriThm selection	USN – 9	Choosing a required algorithmfor specificanalysis	2	High	Gayathri D Divya S Sujashni P Kokila A
Sprint-3	Prediction and analysisof data	USN – 10	Predicting and visualizing the dataeffectively	6	High	Gayathri D Divya S Sujashni P Kokila A
Sprint-4	Report generation	USN – 11	Generating a clear anddetailed reporton productdata analysis	3	High	Sujashni P Kokila A

6.2 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	12	6 days	24 Oct 2022	29 Oct 2022	12	30 Oct 2022
Sprint-2	14	6 days	31 Oct 2022	5 Nov2022	14	6 Nov 2022
Sprint-3	6	6 days	07 Nov2022	12 Nov2022	6	8 Nov 2022
Sprint-4	6	6 days	14 Nov2022	19 Nov2022	6	20 Nov 2022

CHAPTER 7

CODING & SOLUTIONING

7.1 FEATURE

```
from flask import Flask,render_template,request
import cv2
from tensorflow.keras.models import load_model
import numpy as np
from werkzeug.utils import secure_filename

app=Flask(__name__,template_folder="templates")
model=load_model('disaster.h5')
print("Loaded model from disk")

@app.route('/',methods=['GET'])
def home():
    return render_template("index.html")
@app.route('/home',methods=['GET'])
def back():
    return render_template("index.html")
@app.route('/upload',methods=['GET'])
def index():

    cap=cv2.VideoCapture(0)
    H=None
    W= None
```

```
while True:
        (grabbed,frame)= cap.read()

        if not grabbed:
            break
        if W is None or H is None:
            (H,W)= frame.shape[:2]
        output= frame.copy()

        frame=cv2.cvtColor(frame,cv2.COLOR_BGR2RGB)
        frame=cv2.resize(frame,(64,64))
        x=np.expand_dims(frame,axis=0)

        result = np.argmax(model.predict(x),axis=-1)
        index =['Cyclone','Earthquake','Flood','Wildfire']

        output=str(index[result[0]])

        print(result)
        return render_template("output.html",output=output)
        # cv2.putText(output,"activity:{}".format(result),(10,120),cv2.FONT_HERSHEY_PLAIN,1,(0,255,255),1 )
        # cv2.imshow("output",output)
#         if cv2.waitKey(2) & 0xFF==ord('x'):
#             break
# print("[info] cleaning up....")
# cap.release()
# cv2.destroyAllWindows()
```

```
    # return render_template("output.html",output=result)
if __name__=='__main__':
    app.run(host='0.0.0.0',port=8000,debug=False)
```

CHAPTER 8

TESTING

8.1 TEST CASES

Test case ID	Feature Type	Component	Test Scenario	Expected Result	Actual Result	Status
HP_TC_001	UI	Home Page	Verify UI elements inthe HomePage	The Home pagemust be displayed properly	Working as expected	PASS
HP_TC_002	UI	Home Page	Check if the UI elements are displayed properly indifferent screensizes	The Home pagemust be displayed properly in allsizes	The UI is not displayed properlyin screen size 2560 x 1801 and 768 x 630	FAIL
HP_TC_003	Functional	Home Page	Check if thepage redirects to the resultpage once theinput is given	The page shouldredirect to the results page	Working as expected	PASS
BE_TC_001	Functional	Backend	Check if all the routesare working properly	All the routes should properly work	Working as expected	PASS
M_TC_001	Functional	Model	Check if the model can handle various image	The model shouldrescale the imageand predict the	Working as expected	PASS

				results		
M_TC_002	Functional	Model	Check if the modelpredicts the disaster	The model shouldpredict the disaster	Working as expected	PASS
M_TC_003	Functional	Model	Check if the model can handle complexinput	The model should predict the disaster in the	The model failsto identify it since the modelis not built to	FAIL

				compexfeed	handle such data	
RP_TC_001	UI	Result Page	Verify UI elements in the ResultPage	The Result pagemust be displayed properly	Working as expected	PASS
RP_TC_002	UI	Result Page	Check if the resultis displayed properly	The result shouldbe displayed properly	Working as expected	PASS
RP_TC_003	UI	Result Page	Check if the other predictions aredisplayed properly	The other rediction s should be displayed properly	Working as expected	PASS

8.2 USER ACCEPTANCE TESTING

8.2.1 DEFECT ANALYSIS

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Tota l
By Design	1	0	1	0	2
Duplicate	0	0	0	0	0
External	0	0	2	0	2
Fixe d	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

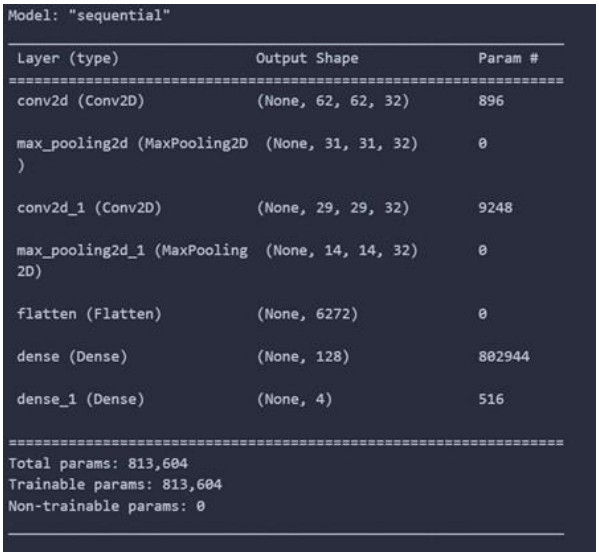
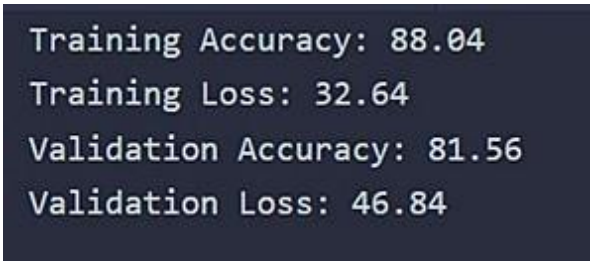
8.2.2 TEST CASE ANALYSIS

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

9.1 PERFORMANCE METRICS

S.No.	Parameter	Values	Screenshot
1.	Model Summary	-	
2.	Accuracy	Training Accuracy – 88.04% Validation Accuracy -81.56%	

CHAPTER 10

ADVANTAGES & DISADVANTAGES

ADVANTAGES:-

1. Humans also need breaks and time offs to balance their work life and personal life. But AI can work endlessly without breaks.
2. With the use of various AI-based techniques, we can also anticipate today's weather and the days ahead.
3. Helpful in getting life back on track..
4. Their Alert nature able to respond effectively and efficiently which defend the society from large scale damages.

DISADVANTAGES:-

1. It involves huge money to be equipped.
2. Problems faced in life basic needs.
3. One application of artificial intelligence is a robot, which is displacing occupations and increasing unemployment .
4. Machines can perform only those tasks which they are designed or programmed to do, anything out of that they tend to crash or give irrelevant outputs which could be a major backdrop.

CHAPTER 11

CONCLUSION

Many researchers have attempted to use different deep learning methodsfor detection of natural disasters. However, the detection of natural disasters by using deep learning techniques still faces various issues due to noise and serious class imbalance problems.To address these problems, we proposed a multilayered deep convolutional neural network for detection and intensity classification of natural disasters. The proposed method works in two blocks—one for detection of natural disaster occurrence and the second block is used to remove imbalanced class issues. The results were calculated as average statistical values: sensitivity, 97.54%; specificity,98.22%; accuracy rate, 99.92%; precision, 97.79%; and F1-score, 97.97% for the proposed model. The proposed model achieved the highest accuracy as compared to other state-of-the-art methods due to its multilayered structure. The proposed model performs significantly better for natural disaster detection and classification, but in the future the model can be used for various natural disaster detection processes.

CHAPTER 12

FUTURE SCOPE

AI -smart technology, which has enabled accurate and speedy solutions. If harnessed properly, the technology has the potential of predicting, preventing andproviding response faster than ever.

AI data setups are trained to predict seismic data to analyze the patterns of earthquake occurrences, rainfall records and monitor flooding, measure the intensity of hurricanes and read the geological data to understand volcanic eruptions, such systemscan reduce the catastrophic impact of natural disasters.

Last year, Google's Pilot project to monitor flood in India with the help of AI, wasa successful one – it was a Patna project. They were able to predict floods and the regions that it would be affected due to the natural disaster with an accuracy of over 90%. It was possible owing to the combination of data from government agencies that provide on-ground information – from measuring devices placed on the spot and satellite captured images of flood-prone areas. They ran hundreds of thousands of simulations on its machine learning (ML) models to predict the flow of water.In the future, leveraging AI can help disaster management bodies install drones, sensors androbots to provide accurate information about damaged buildings and landscapes, potential floods, making rescue missions safer and less time-consuming.

There is a need for smart technology to be integrated within our local communities. Immediate response and tech-based solutions can help reduce the extentof damage. However, since AI is based on machine codes, there is a scope of limitations and errors. However, the amalgamation of human, empathy and alertness, could do wonders in the field of crisis management.

APPENDIX

SOURCE CODE

MODEL CREATION

```
import numpy as np
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Flatten, Conv2D, MaxPooling2D
from keras.preprocessing.image import ImageDataGenerator
```

```
data_path = './Dataset/'
batch_size = 32
target_size = (64, 64)
```

```
train_datagen = ImageDataGenerator(rescale=1./255,
                                   shear_range=0.2,
                                   zoom_range=0.2,
                                   width_shift_range=0.1,
                                   height_shift_range=0.1,
                                   horizontal_flip=True,
                                   validation_split=0.2)

test_datagen = ImageDataGenerator(rescale=1. / 255, validation_split=0.2)
```

```
X_train = train_datagen.flow_from_directory(data_path,
                                            target_size=target_size,
                                            batch_size=batch_size,
                                            color_mode="rgb",
                                            subset="training",
                                            class_mode='categorical')


X_test = test_datagen.flow_from_directory(data_path,
                                          target_size=target_size,
                                          batch_size=batch_size,
                                          color_mode="rgb",
                                          subset="validation",
                                          class_mode='categorical')
```

```
Found 3544 images belonging to 4 classes.
Found 884 images belonging to 4 classes.
```

```
model = Sequential()
```

```
    model.add(Conv2D(32, (3, 3), input_shape=(64, 64, 3), activation='relu'))
    model.add(MaxPooling2D(pool_size=(2, 2)))

    model.add(Conv2D(32, (3, 3), activation='relu'))
    model.add(MaxPooling2D(pool_size=(2, 2)))

    model.add(Flatten())

    model.add(Dense(units=128, activation='relu'))
    model.add(Dense(units=4, activation='softmax'))
```

```
    model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 62, 62, 32)        896

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

```
 conv2d_1 (Conv2D)           (None, 29, 29, 32)        9248

 max_pooling2d_1 (MaxPooling  (None, 14, 14, 32)       0
 2D)

 flatten (Flatten)           (None, 6272)              0

 dense (Dense)               (None, 128)               802944

 dense_1 (Dense)             (None, 4)                 516

=================================================================
Total params: 813,604
Trainable params: 813,604
Non-trainable params: 0
_________________________________________________________________
```

```
    model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

```
model.fit(X_train,
          steps_per_epoch=len(X_train),
          epochs=20,
          validation_data=X_test,
          validation_steps=len(X_test))
```

```
Output exceeds the size limit. Open the full output data in a text editor
Epoch 1/20
111/111 [==============================] - 63s 495ms/step - loss: 1.0808 - accuracy: 0.5282 - val_loss: 0.7762 - val_accuracy: 0.7025
Epoch 2/20
111/111 [==============================] - 52s 470ms/step - loss: 0.7012 - accuracy: 0.7322 - val_loss: 0.7050 - val_accuracy: 0.7353
Epoch 3/20
111/111 [==============================] - 50s 448ms/step - loss: 0.6575 - accuracy: 0.7525 - val_loss: 1.1470 - val_accuracy: 0.6210
Epoch 4/20
111/111 [==============================] - 47s 428ms/step - loss: 0.5846 - accuracy: 0.7847 - val_loss: 0.8129 - val_accuracy: 0.6980
Epoch 5/20
111/111 [==============================] - 49s 442ms/step - loss: 0.5523 - accuracy: 0.7999 - val_loss: 0.6012 - val_accuracy: 0.7760
Epoch 6/20
111/111 [==============================] - 52s 470ms/step - loss: 0.5448 - accuracy: 0.7923 - val_loss: 0.7817 - val_accuracy: 0.7048
Epoch 7/20
111/111 [==============================] - 46s 415ms/step - loss: 0.4935 - accuracy: 0.8149 - val_loss: 0.6035 - val_accuracy: 0.7602
Epoch 8/20
111/111 [==============================] - 45s 406ms/step - loss: 0.4554 - accuracy: 0.8361 - val_loss: 0.5008 - val_accuracy: 0.8111
```

```
Epoch 9/20
111/111 [==============================] - 45s 405ms/step - loss: 0.4598 - accuracy: 0.8335 - val_loss: 0.6498 - val_accuracy: 0.7557
Epoch 10/20
111/111 [==============================] - 45s 403ms/step - loss: 0.4260 - accuracy: 0.8420 - val_loss: 0.7311 - val_accuracy: 0.7217
Epoch 11/20
111/111 [==============================] - 45s 406ms/step - loss: 0.4475 - accuracy: 0.8352 - val_loss: 0.4500 - val_accuracy: 0.8224
Epoch 12/20
111/111 [==============================] - 45s 403ms/step - loss: 0.4096 - accuracy: 0.8507 - val_loss: 0.5084 - val_accuracy: 0.7952
Epoch 13/20
...
Epoch 19/20
111/111 [==============================] - 43s 387ms/step - loss: 0.3500 - accuracy: 0.8685 - val_loss: 0.4097 - val_accuracy: 0.8473
Epoch 20/20
111/111 [==============================] - 44s 392ms/step - loss: 0.3264 - accuracy: 0.8804 - val_loss: 0.4684 - val_accuracy: 0.8156

<keras.callbacks.History at 0x19d4c8a69b0>
```

```
model.save("disaster.h5")
```

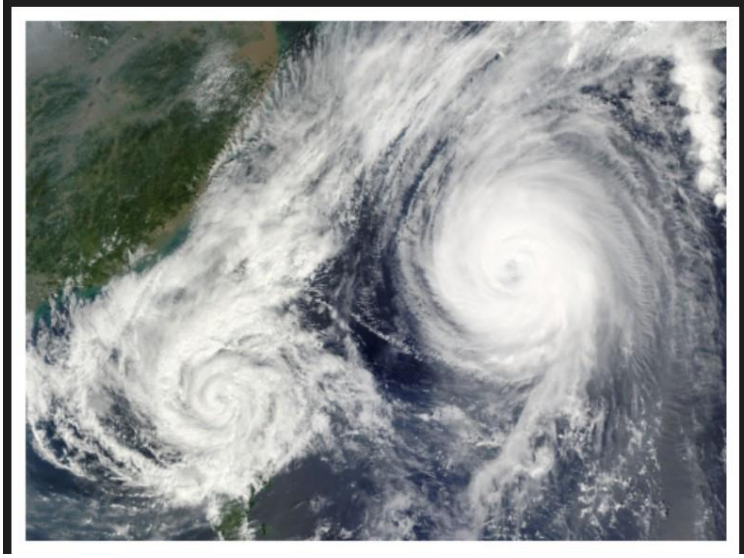
```python
from tensorflow.keras.preprocessing import image
import matplotlib.pyplot as plt

def predict(image_path):
    img = image.load_img(image_path, target_size=target_size)
    x = image.img_to_array(img)
    x = np.expand_dims(x, axis=0)

    labels = ['Cyclone', 'Earthquake', 'Flood', 'Wildfire']

    pred = model.predict(x)
    prediction = labels[np.argmax(pred[0])]

    print(f'Disaster: {prediction}')
    plt.imshow(plt.imread(image_path))
    plt.axis('off')
    plt.show()
```

```python
predict('./Sample/1.jpg')
```

```
1/1 [==============================] - 0s 26ms/step
Disaster: Wildfire
```

```
predict('./Sample/2.jpg')
```

```
1/1 [==============================] - 0s 26ms/step
Disaster: Cyclone
```



```
predict('./Sample/3.jpg')
```

```
1/1 [==============================] - 0s 31ms/step
Disaster: Earthquake
```

```
predict('./Sample/4.jpg')
```

```
1/1 [==============================] - 0s 26ms/step
Disaster: Flood
```

FLASK APP.PY

```
from flask import Flask,render_template,request
import cv2
from tensorflow.keras.models import load_model
import numpy as np
from werkzeug.utils import secure_filename

app=Flask(__name__,template_folder="templates")
model=load_model('disaster.h5')
print("Loaded model from disk")

@app.route('/',methods=['GET'])
def home():
    return render_template("index.html")
@app.route('/home',methods=['GET'])
def back():
    return render_template("index.html")
@app.route('/upload',methods=['GET'])
def index():

    cap=cv2.VideoCapture(0)
    H=None
    W= None
    while True:
            (grabbed,frame)= cap.read()

            if not grabbed:
                break
            if W is None or H is None:
                (H,W)= frame.shape[:2]
```

```
            output= frame.copy()

            frame=cv2.cvtColor(frame,cv2.COLOR_BGR2RGB)
            frame=cv2.resize(frame,(64,64))
            x=np.expand_dims(frame,axis=0)

            result = np.argmax(model.predict(x),axis=-1)
            index =['Cyclone','Earthquake','Flood','Wildfire']

            output=str(index[result[0]])

            print(result)
            return render_template("output.html",output=output)
            # cv2.putText(output,"activity:{}".format(result),(10,120),cv2.FONT_HERSHEY_PLAIN,1,(0,255,255),1 )
            # cv2.imshow("output",output)
    #         if cv2.waitKey(2) & 0xFF==ord('x'):
    #             break
    # print("[info] cleaning up....")
    # cap.release()
    # cv2.destroyAllWindows()

    # return render_template("output.html",output=result)
if __name__=='__main__':
    app.run(host='0.0.0.0',port=8000,debug=False)
```

INDEX . HTML

```
<html>
    <head>

       <meta charset="utf-8" name="viewport" content= "width=device-width, initial-scale=1.0">
      <style>
          *{
           margin: 0;
           padding: 0;
          }
          body{
           width: 100vw;
           height: 100vh;
          }
          .content{
           display: flex;
           flex-direction: column;
           margin-top: 5rem;

          }

          .top{
              height: 3rem;
              background-color: rgb(12, 104, 161);
              color: white;
              display: flex;
              letter-spacing: 2px;
              align-items: center;
              font-weight: 900;
              font-size: 1.2rem;
```

```
            font-family: monospace;
            padding-left: 2.5rem;
        }
        .images{
            display: flex;
            justify-content: center;
            align-items: center;

        }
        img{

            width:200px;
            height:300px ;
            border-radius: 1rem;
            margin-left: 5rem;
        }
        button{
            position: relative;
            background-color: rgb(12, 104, 161);
            color: white;
            padding: 5px 10px;
            border: 1px solid  rgb(12, 104, 161);
            font-weight: 900;
            font-size: 1.2rem;
            font-family: monospace;
            display: flex;
            margin-left: 46%;
            margin-top: 5rem;
        }
```

```
    </style>
</head>
    <body>


        <div class="top">
            <div id="heading">AI Based Natural Disaster Analysis </div>
        </div>
        <div class="content">
        <div class="images">
                <img src="../static/images/cyclones.jpg">
                <img src="../static/images/earthquake.jpg">
                <img src="../static/images/flood.jpg" >
                <img src="../static/images/wildfire.jpg">
        </div>
        <form action="/upload" method="get">
            <button type="submit">Open Web Camera</button>
        </form>
      </div>



    </body>
</html>
```

OUTPUT . HTML

```
<html>
    <head>
        <style>
            *{
                padding: 0;
                margin: 0;
            }
            body{
                height: 100vh;
                width: 100vw;
            }


           .top{
                height: 3rem;
                background-color: rgb(12, 104, 161);
                color: white;
                display: flex;
                letter-spacing: 2px;
                align-items: center;
                font-weight: 900;
                font-size: 1.2rem;
                font-family: monospace;
                padding-left: 2.5rem;
            }
            .top #home{
                position:absolute;
                background-color: rgb(12, 104, 161);
                color: white;
```

```
                right: 40;
                padding: 2px 8px;
                font-size: 1rem;
                font-family: monospace;
                border: 1px solid rgb(255, 255, 255);
                border-radius: 5px;

            }
            .content{

                display: flex;
                justify-content: center;

            }
             #name{
                padding: 3rem;
                text-transform: uppercase;
                font-size:2rem ;
                font-weight: 900;
                text-align: center;
                font-family:'Times New Roman', Times, serif;
            }
           img{

                width:400px;
                height:400px ;
                border-radius: 4rem;
            }
```

```
        </style>
    </head>
    <body>

        <div class="top">
         <div id="heading">AI Based Natural Disaster Analysis </div>
         <form action="/home" method="get">
             <button type="submit" id="home">Home</button>>
         </form>

        </div>

        <div id="name">{{output}}</div>
        <div class="content">
        <img src="../static/images/emergency.png" alt="">

         </div>
    </body>
</html>
```

GITHUB:-

https://github.com/IBM-EPBL/IBM-Project-41075-1660639176

PROJECT DEMO:-

https://youtu.be/G4izm6_VOL8