Project Report

TEAM ID : PNT2022TMID00473

INTRODUCTION 1.1

Project Overview:

Machine learning algorithms can be used by businesses to as accurately predict changes in consumer demand as feasible. These algorithms are capable of automatically recognising patterns, locating intricate links in big datasets, and picking up indications for changing demand. A food delivery service has to deal with a lot of perishable raw materials which makes it all, the most important factor for such a company is to accurately forecast daily and weekly demand. Too much inventory in the warehouse means more risk of wastage, and not enough could lead to out-ofstocks - and push customers to seek solutions from your competitors. The replenishment of majority of raw materials is done on weekly basis and since the raw material is perishable, the procurement planning is of utmost importance, the task is to predict the demand for the next 10 weeks 1.2 .

Purpose:

The main aim of this project is to create an appropriate machine learning model to forecast the number of orders to gather raw materials for next ten weeks. To achieve this, we should know the information about of fulfilment center like area, city etc., and meal information like category of food sub category of food price of the food or discount in particular week. By using this data, we can use any classification algorithm to forecast the quantity for 10 weeks. A web application is built which is integrated with the model built.

2. LITERATURE SURVEY

2.1 Existing problem

Frequent fires in the Himalayan region of Uttaranchal in the Indian Himalayas have been blamed for forest deterioration. It is true that frequent fires on large scales cause air pollution, mar quality of stream water, threaten biodiversity and spoil the aesthetics of an area, but fire plays an important role in forest ecosystem dynamics. Moreover, it is not fire, but other anthropogenic activities plus fire that are degrading the forest of the Indian Himalayas. In the present study the role of fire in shaping forest structure and composition is analysed. If fire is managed wisely it can be used as the cheapest means of forest management. For this purpose different fire characteristics are assessed together with their interrelationship with forest flora.

2.2 References:

[1] In fire detection, the color of the image from a camera is highly important. Sometimes, it does not possible to watch the entire forest images according to the size as it may be some difficulties in detecting the fire.So that, using Convolutional Neural Network(CNN) technology would be easier to avoid the blindness and accurate level of fire identification. It uses the support vector mechanism for the image classification.In this technique, the image is segmented based on the color of the flame and transferred to the CNN network. This would be

found out more attributes and decide there is a fire occurs or not. Fire can be detected by analyzing the color of the flame in a picture. Finding the fire by using the number of pixels plotted in a picture according to the fire color and can be measure the intensity of the fire. So that, it should be easier to detect fire and stamp out the fire. The system should be trained and tested using a large amount of data. Algorithms are used for the segmentation of images and in finding the fire. This method should be more effective and reliable in identifying the fire. The accuracy should be much better than the other methods.(Yuanbin Wang, 2019)

[2] In this paper, Environment can be destroyed by the forest fire, and it could be making a huge amount of loss. Recently, the amazon forest has had a fire and it remained for over 15 days. This resulted a huge loss and it affected negatively to the diversity and global conditions. The wireless sensor networks help in detecting the forest fire.Itcan give a warning as soon as if there any unusual event occurs. Sometimes, these networks can be making false alarms according to the wrong detection. In such cases, machine learning mechanism can be used to prevent such cases. Earlier, satellite-based systems are used to detecting fire. But it may not be possible to finding the distraction as it took pictures of surface of the earth in every two days. As a result, it may not be considered as an effective method. Also, the weather conditions may be affected in the quality of the pictures. Another method for the fire detection was using watch towers. It was handled manually by watching the whole forest area in a tower and finding if there any fire occurs. Another one is using optical sensors and digital camera. It would not be much effective as the vision can be distracted by the high trees or hills.

[3] In this paper, a novel method for fire detection is proposed based on ensemble learning. The dataset is created using 10581 images from various public sources like BowFire , FD-Dataset , ForestryImages , VisFire . The dataset is preprocessed and fed into not just one but two individual object detectors, YOLOv5 and EfficientDet integrated in parallel mode to achieve better accuracy than a single object detector. Although it uses integrated object detectors, this does not take the whole image into consideration. Therefore, another classifier is introduced to solve this problem. EfficientNet takes the image as whole and evaluates the image to enable total advantage of the information. The results will be decided by a decision strategy algorithm which takes the opinion of the three individual object detectors into account which in turn improves the performance of the model and decrease the rate of False positives. This paper claims that they have achieved a superior tradeoff average accuracy, average recall, false positive and latency.

[4] This paper proposes that forest fires can be detected by vision-based fire detection systems which can be mounted to an unmanned aerial vehicle (UAVs) for strategically scanning acreage of fire prone areas. This paper also strongly recommends Convolutional neural networks for identifying smoke and fire through videoframes which is taken as images. They have collected the dataset from different internet sources. They have resized the images to canonical size of

240x320. In this paper, the basic idea is to find the fire patches in an image. The authors propose two methods for the algorithm to build the model. First was to apply fire patch classifier from scratch. Second was to teach a full image classifier and apply fine-tuned patch classifier if the image contains fire. Then they compare SVM-pool5 (Support vector machines) with CNN-pool5, the accuracies recorded are 95.6% and 97.3% respectively with a detection rate of 84.8%, making CNNpool5 network more accurate than SVM-pool5 classifier.

[5] In this paper, the author uses CNN-convolutional neural networks to detect fire with the help of live video footage through anti-fire surveillance systems. The paper proposes YOLOv2 convolutional neural network is one of the best solutions for detecting fire and smoke both indoor and outdoor environment. You only look once (YOLO) is a deep learning model for object detection, YOLOv2 is the next version which has been upgraded to rectify the setbacks of YOLO namely the inaccuracy to locate and mark the region of interest in the images and the lower recall rate compared to other region-oriented algorithms. Thus, increasing the efficiency of the architecture. They started with an input image of size 128x128x3. They used convolutional layers to map the features on the input image. The features extracted are then given as input to YOLOv2 object detection subnetwork. YOLOv2 Transform layer is implemented to improve network stability for object localization.

[6] This research paper, the authors propose a cost-effective fire detection using CNN from surveillance videos. This papers critically analyses the statistics of deaths due to fire. So, their focus is to propose a system that is home friendly and commercial. This paper gives us an insight of how to carefully select the data properly, how to analyse the computational complexity and detection accuracy. They use a model called GoogleNet for extracting the features from the images. For reducing the complexity of larger patches, they reduce dimensionality. The model is tested with two different datasets for validation purposes and results are compared. They achieved an accuracy of 93.5% on the first dataset and an 86% on the next dataset.

[7] Deep learning and wireless sensor network can be helpful in forest fire detection. The research put forward a system using these approaches can detect the forest fire in the early stages. Using the deep learning model, the system detects the fire according to the collection of data from different sensor networks placed widely in the forest. Here, the system consists of the Internet of Things used as a main concept, moving or fixed sensors and a suitable deep learning model. More accurately, there are several sensor nodes places within each 1 km distance and these nodes are transfer data to the internet servers through the gateways. Then this collected information is displayed in a dashboard with online network. Each nodemeasures the values of humidity, carbon monoxide, temperature, carbon dioxide, and atmospheric pressure. These factors have a major role in the forest fire. In this method, firstly, it calculates the weather information from the weather detector located in forest and then find out the Fire weather index

(FWI) using the sensor nodes with the help of deep learning algorithms and the metrics. If the FWI have value changes, the Unmanned Aerial Vehicle (UAV) helps to detect these sensor values more accurately to find the existence of fire. Also, the control tower act as a fire distinguisher to distinguish the fire (Wiame Benzekri1, 2020).

[8] Fire can be detected by using the amount of smoke. The smoke sensors are used to measure the amount of smoke from the fire, and it could be compared with a threshold value and if it is beyond that value, it is considered as a fire scenario.Using image processing, fire can be detected as soon as possible. Fixing the CCTV camera everywhere and the images from these cameras can be processed to monitor the fire. If any changes occur, it is easy to detect and extinguish the fire quickly. This system has a water extinguisher for extinguish the fire when the alarm turns on. The CCTV camera is used for recording the video of a particular spot and it is connected to a mini- computer called Raspberry-pi. So that it could get the constant video recording of a particular area. The captured video pictures are processed frame by frame and once the fire detected, the alarm would be turn on. Also, the alarm would be turned off when the fire extinguished completely.The Virtual Network Computing is used for the execution of the program, where the details of video are transferred from the raspberry-pi to the viewing computer.This system includes detection, alert, fire extinguish, software and network modules.

[9] A research study proposes a system which is a combination of using neural networks, computer vision rules, and other expert rules helps in detecting the forest fire. Different approaches are applied to build this system; visual infrared image matching, using the previous hazards memory, image processing, location, size, and geographical data. Here, infrared cameras, visual cameras, meteorological sensors are using for the collection of input data. The image processing tool is combined with the visual and infrared processing. Infrared processing is a combination of detection, oscillation, and alarm processing processes. The growing-region algorithm is used to separate the false alarms. The visual processing finds out the exact location of the visual image from the infrared analysing process. By using different algorithms, it can be detected and easily reject the false alarms. The meteorological information used to detect the humidity, temperature and other factors which affect the forest fire. So that, it is easy to estimate the possibility of fire. Using this proposed system, it can be detecting the forest fire in early stage and avoid the false detection (Begoña C. Arrue, 2000).

[10] This paper put forward an approach in real-time forest fire detection using wireless sensor network paradigm. This method can detect and forecast the fire more accurately than the other methods used in forest fire detection. Firstly, the sensor networks acquire the details about the humidity, smoke, temperature, and wind speed as these factors affect the forest fire. The sensor nodes are placed widely in the forest, and it is arranged into clusters. The sensor nodes use GPS to track their location as they can sends these location details along with the data such

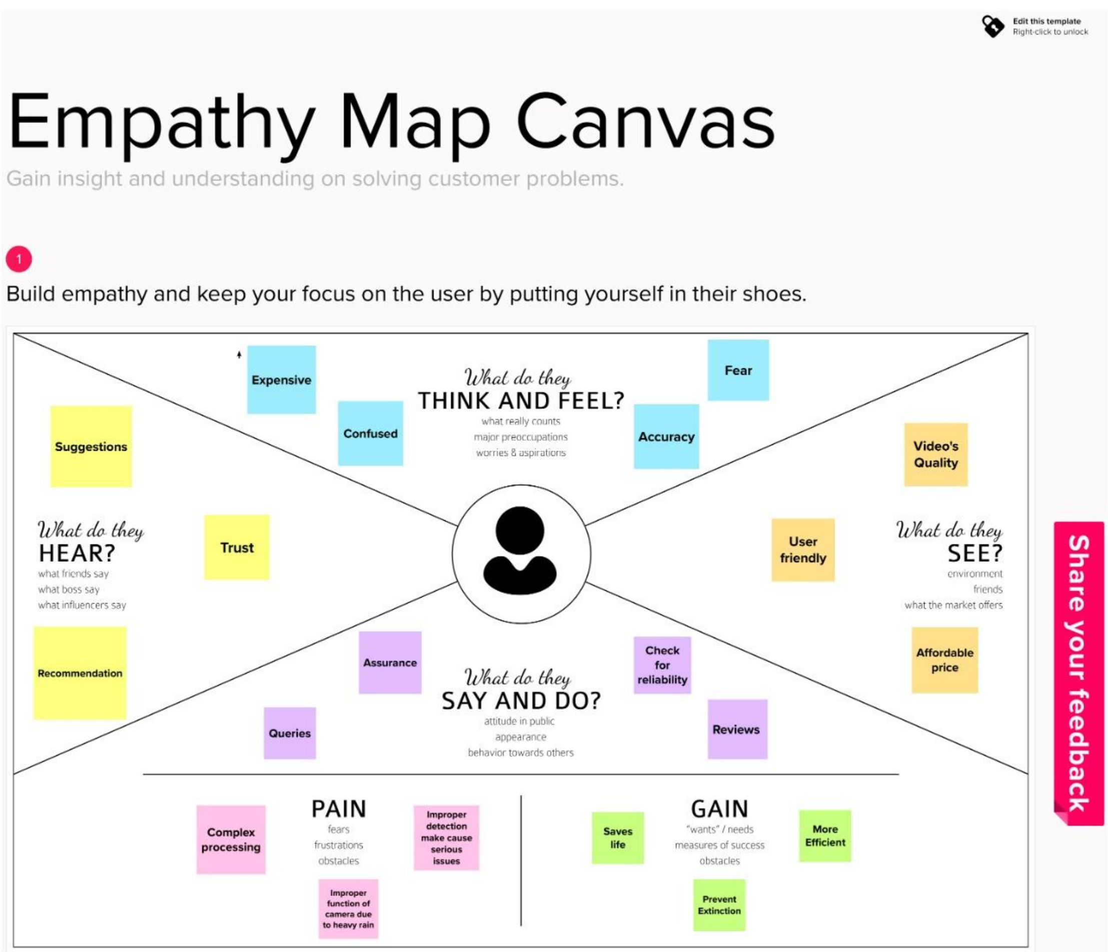
as measurements of temperature to the cluster head. Then, using a neural network method, the cluster header computes the weather index and then these information sends to the manager node. The wind speed is calculated by the wind sensor nodes, which are manually placed in the forest. The users get information from the manager node when an abnormal event occurs like high temperature and smoke. As well as manager node gives information about the levels of forest fire risk rate according to the weather index from different clusters. So that, users can easily find out the exact location of fire in the forest if it occurs. Also, they could protect the forest from the fire hazard due to the early detection (Liyang Yu, 2005).

2.3 Problem Statement Definition

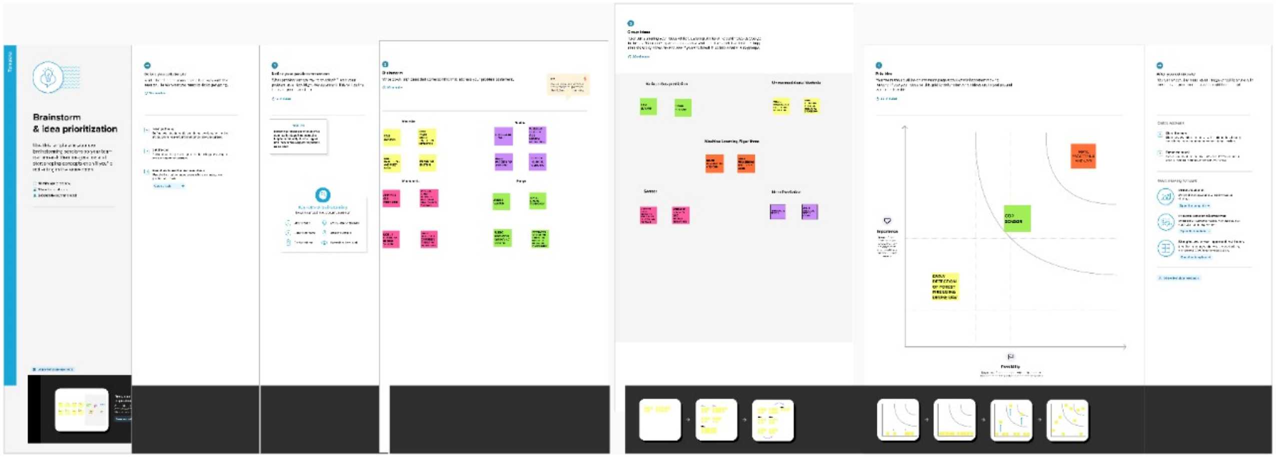
This problem statement aims to detect the presence of Forest Fire at an early stage .This project is indented to design a forest fre prediction mechanism. Forest fre prediction scheme based on image processing ,Deep learning has been proposed for the prediction in the timely manner .The proposed project is helpful to identify the fre signals and inform the respective person as an alert to take an appropriate action to prevent the forest fre and save the many trees and animals.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming



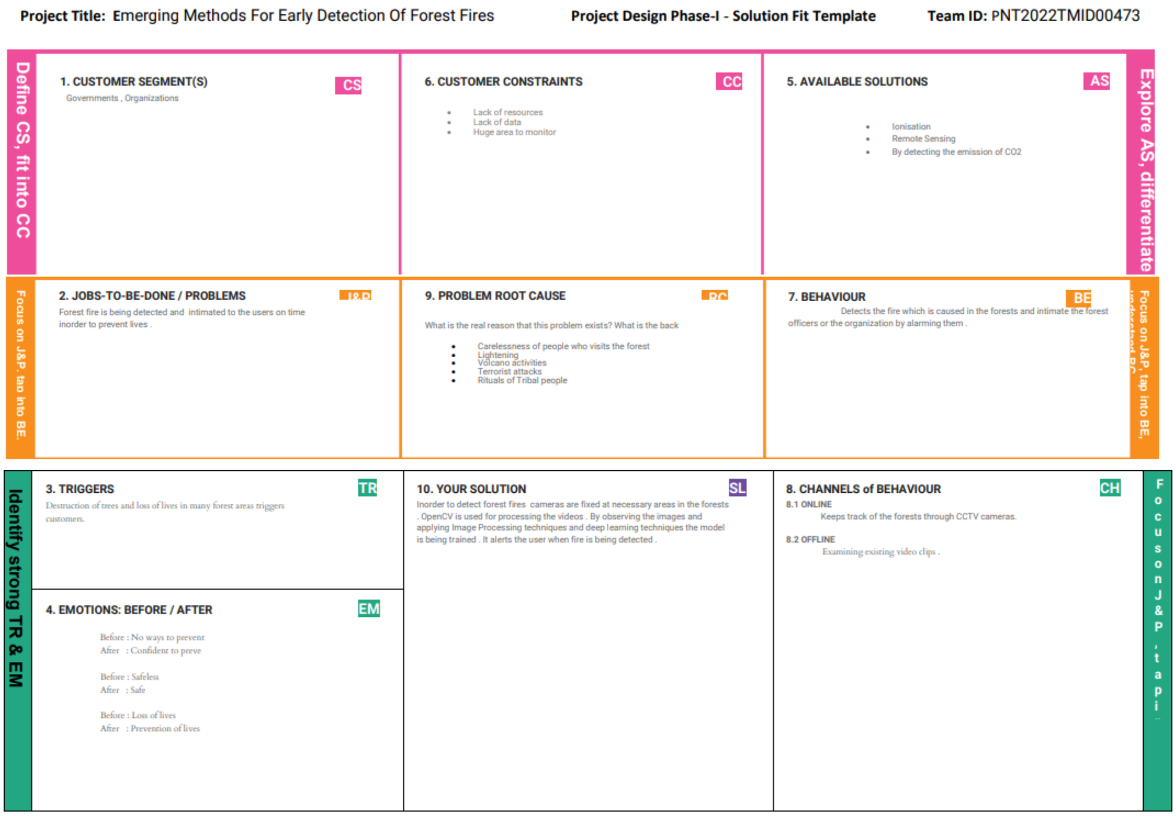
3.3 Proposed Solution

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

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To build a Deep Learning Model which can detect the forest fires. The objective for such a model is that it can be used to prevent lives in Forest. It gives accurate results. The model is being trained with training datasets in order to detect forest fires. Forest Fire Detection is based on Image Processing, Deep Learning has been proposed for the prediction in a timely manner. The proposed project is helpful to identify the fire signals and inform the respective person as an alert to take an appropriate action to prevent the forest fires.

2.	Idea / Solution description	To accomplish this ,the first step is to collect the dataset or create the dataset. The second step is image preprocessing in which the imagedatagenerator library is being imported and it's functionality is being applied to Trainset and Testset. The third step is Model Building in which the model building Libraries are being imported. Add CNN layers ,Hidden Layer and Output Layer . Then Configure the Learning Process . After training and testing the model optimize and save the model. The fourth step is Video Streaming and Alerting . Use OpenCV for processing the video . Create an account in Twilio service and use Twilio API to send messages .
3.	Novelty / Uniqueness	▪ Deep Learning Based Fire Detection. ▪ Image processing
4.	Social Impact / Customer Satisfaction	Customer (People Near Forest, Government) need not required to detect it manually. Early Detection is Better than Damage.

3.4 Problem Solution fit



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	SubRequirement (Story / Sub-Task)
FR-1	User Registration	Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	Login using credentials
FR-4	User Search	Search for Info on forest fire occurrence
FR-5	User Profile	User shall be given a live feed of the forest
FR-6	User Application	User is alerted if there is a forest fire occurrence in their surroundings

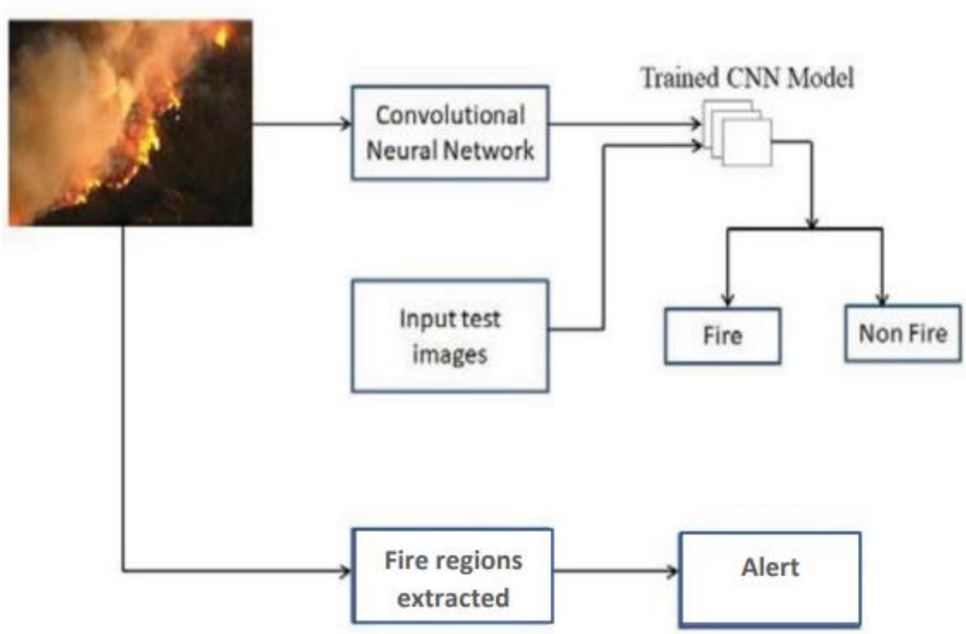
4.2 Non-Functional requirements

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Alerts according to the user location
NFR-2	**Security**	Instant live feed with alert of the situation
NFR-3	**Reliability**	The prediction of the forest fire is 87% accurate
NFR-4	**Performance**	The feed and the alert message an immediate action without a lag
NFR-5	**Availability**	The application gives alerts and live feeds24/7
NFR-6	**Scalability**	Early detection and alerting users are done efficiently and in a faster means

5. PROJECT DESIGN

5.1 Data Flow Diagrams

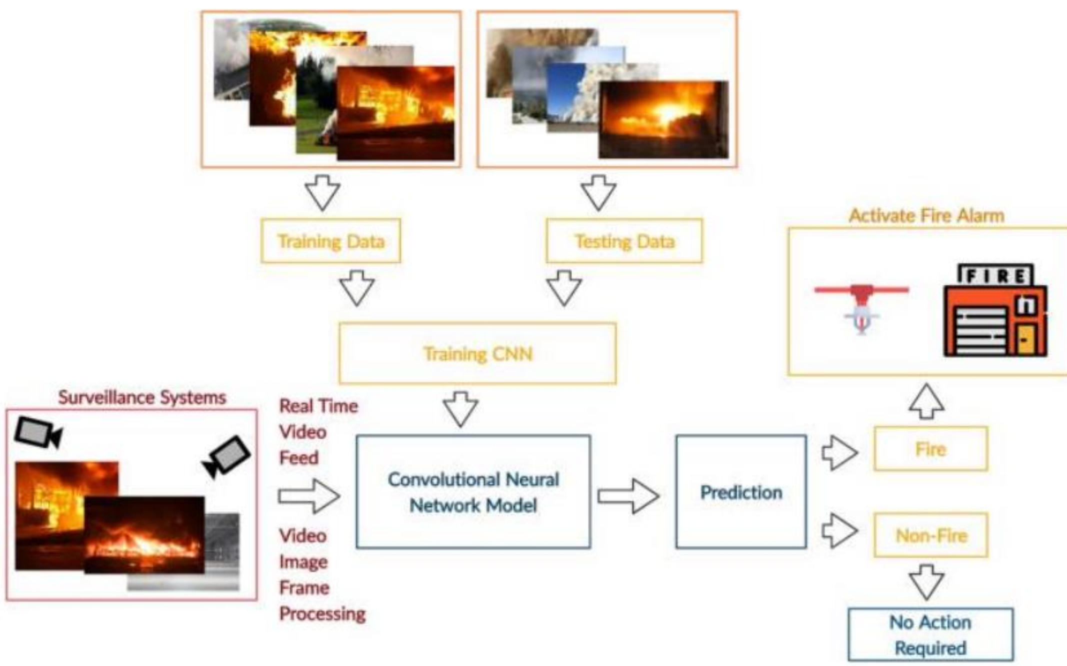
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



Data Flow Diagram- Emerging Methods For Early Detection Of Forest Fires

1. COLLECT DATA
2. EVALUATE DATASET
3. IMPLEMENT ALGORITHMS
4. EVALUATE THE ACCURACY OF EACHALGORITHMS
5. DISPLAY RESULTS

5.2 Solution & Technical Architecture



Architecture and data flow of forest fire detection

5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Environmentalist	Collect the data	USN-1	As an Environmentalist , it is necessary to collect the data of the forest which includes temperature , humidity , wind	It is necessary to collect the right data else the prediction may become wrong	High	Sprint-1

			and rain of the forest			
		USN-2	Identify algorithms that can be used for prediction	To collect the algorithm to identify the accuracy level of each algorithm	Medium	Sprint-2
		USN-3	Identify the accuracy of each algorithms	Accuracy of each algorithm calculated so that it is easy to obtain the most accurate output	High	Sprint-2
		USN-4	Evaluate the Dataset	Data is evaluated before processing	Medium	Sprint-1
		USN-5	Identify accuracy , precision , recall of each algorithms	These values are important for obtaining the right output	High	Sprint-3
		USN-6	Outputs from each algorithm are obtained	It is highly used to predict the effect and to take precautionary measures	High	Sprint-4

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Name	Milestone Number	Description	Mandatory	Optional
Project Objectives	M-001	We will be able to learn to prepare dataset, image processing, working with CNN layers, read images using OpenCV and CNN for computer vision AI	Yes	-
Project Flow	M-002	A project management process flowchart is a graphical aid, designed to visualize the sequence of steps to be followed throughout the project management process	Yes	
Pre-Requisites	M-003	To complete this project we should have known following project such as Keras, Tensorflow, Python , Anaconda, OpenCV, Flask, Scikit-learn etc…	Yes	
Prior Knowledge	M-004	One should have knowledge on the Supervised Learning ,CNN and Regression Classification and Clustering, ANN	Yes	

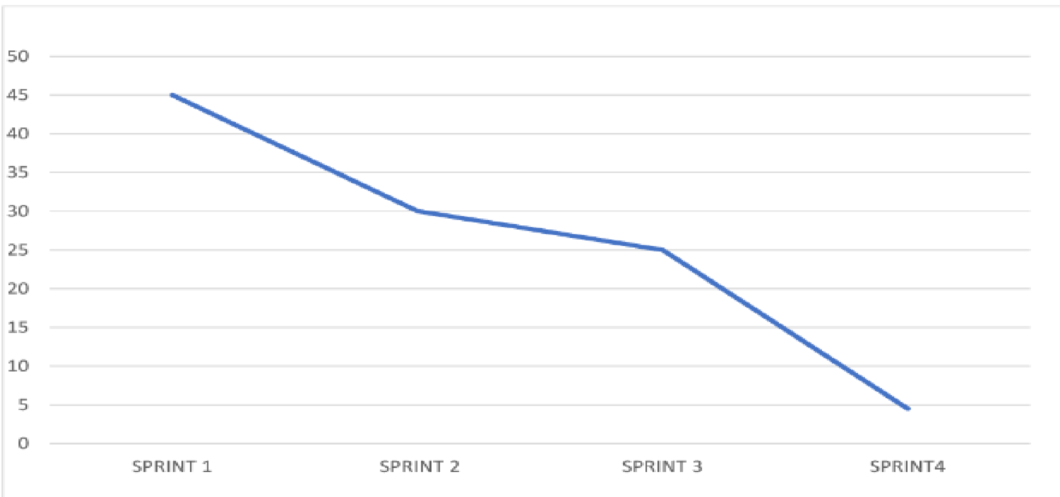
Data collection	M-005	We can collect dataset from different open sources like kaggle.com, UCI machine learning etc	Yes	
Image Preprocessing	M-006	Importing the ImageDataGenerator libraries, Define Parameters/Arguments for ImageDataGenerator class, Applying Image Data Generator Functionality to trainset and testset	Yes	
Model Building	M-007	Importing the model building libraries, Initializing the model, Adding CNN layers, Adding Dense layers, Configuring the learning Process ,Train the model, Save the model, Predictions.	Yes	
Video Analysis	M-008	Open cv for video processing, creating an account in twilio service and Sending alert messages.	Yes	
Train CNN model	M-009	Register for IBM Cloud and train Image Classification Model	Yes	
Ideation Phase	M-010	Prepare Literature Survey on the selected Project and Information Gathering, empathy map and ideation	Yes	
Project Design Phase-I	M-011	Prepare Proposed solution , problem-solution fit and Solution Architecture	Yes	
Project Design Phase-II	M-012	Prepare Customer journey ,functional requirements, Data flow diagram and Technology Architecture	Yes	
Project Planning Phase	M-013	Prepare Milestone list , Activity list and Sprint Delivery Plan	Yes	
Project Development Phase	M-014	Project Development delivery of Sprint 1, Sprint 2, Sprint 3, Sprint 4	Yes	

6.2 Sprint Delivery Schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story /Task	Story Points	Pr
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	20	Hi
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application usage.	20	Hi
Sprint-2	Input	USN-3	Whenever the fire is detected, the information is given to the database.	20	Hi
Sprint-2		USN-4	When it is the wildfire then the alarming system is activated.	20	Hi

Sprint-3	Output	USN-5	And the alarm also sent to the corresponding departments and made them know that the wildfire is erupted	20	Hi
Sprint-4	Action	USN-6	Required actions will be taken in order to control erupted wildfire by reaching as early as possible to the destination with the help of detecting systems.	20	Hi

6.3 Reports from JIRA



7. CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 Feature 1

```
[ ] train_dataset = train.flow_from_directory("/content/drive/MyDrive/Dataset/train_set",
                                               target_size=(128,128),
                                               batch_size = 32,
                                               class_mode = 'binary' )

    Found 443 images belonging to 2 classes.
```

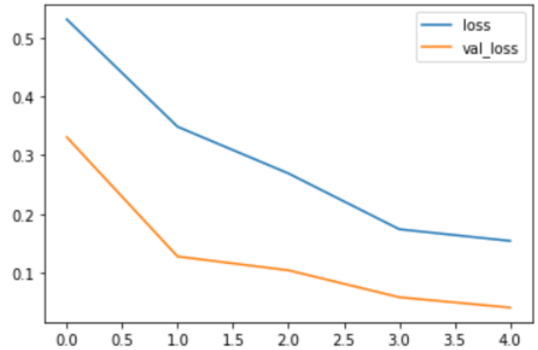
```
test_dataset = test.flow_from_directory("/content/drive/MyDrive/Dataset/test_set",
                                         target_size=(128,128),
                                         batch_size = 32,
                                         class_mode = 'binary' )

Found 121 images belonging to 2 classes.
```

7.2 Feature 2

```
[ ] model = load_model("/content/forest1.h5")
```

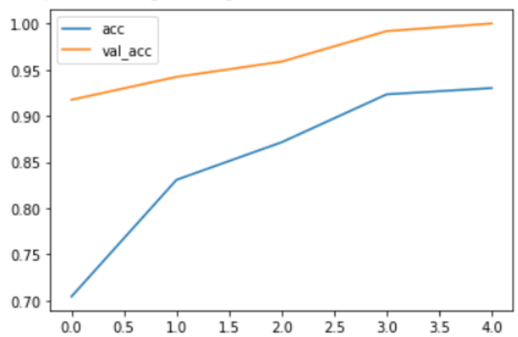
```
[ ] import matplotlib.pyplot as plt
    plt.plot(r.history['loss'],label='loss')
    plt.plot(r.history['val_loss'],label='val_loss')
    plt.legend()
```

```
<matplotlib.legend.Legend at 0x7f94a329e810>
```




```
[ ] plt.plot(r.history['accuracy'],label='acc')
    plt.plot(r.history['val_accuracy'],label='val_acc')
    plt.legend()
```

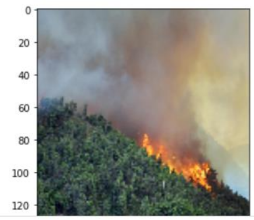
```
<matplotlib.legend.Legend at 0x7f94a32a7910>
```



```
[ ] def predictImage(filename):
      img1=image.load_img(filename,target_size=(128,128))
      plt.imshow(img1)
      y=image.img_to_array(img1)
      x=np.expand_dims(y,axis=0)
      val=model.predict(x)
      print(val)
      if val==0:
        plt.xlabel(" NO fire",fontsize=30)
      elif val==1:
        plt.xlabel("fire",fontsize=30)
```

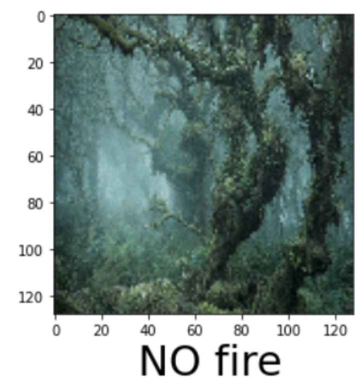
```
[ ] predictImage("/content/drive/MyDrive/Dataset/test_set/with fire/Forest_Fire (3).jpg")
```

```
1/1 [==============================] - 0s 90ms/step
[[1.]]
```



```
[ ] predictImage("/content/drive/MyDrive/Dataset/test_set/forest/01_NeilBurnell_Mystical_photoverticall.jpg")
```

```
1/1 [==============================] - 0s 36ms/step
[[0.]]
```



8. TESTING

8.1 Test Cases

By the showing image of forest fire the desired output of “Fire detected “ is sent and alerted by an audio.

```
1/1 [==============================] - 0s 100ms/step
[[1.]]
SM0df87da05565a3b5c343504f724bd82b
Fire detected
SMS sent
```

0:00 / 1:24

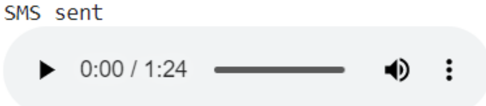
8.2 User Acceptance Testing

We have tested our project by showing the image of forest with fire and forest without fire.The output is shown above.

9. RESULTS

9.1 Performance Metrics

```
1/1 [==============================] - 0s 100ms/step
[[1.]]
SM0df87da05565a3b5c343504f724bd82b
Fire detected
SMS sent
```



10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- Avoid Smoke Inhalation. The most important reason is perhaps the only one you really need.
- Early Detection. The earlier a fire is detected, the faster it will be that firefighters will respond.
- Insurance Discounts.
- 24/7 Monitoring.
- Easy & Affordable

DISADVANTAGES

- The system is essentially useless if the batteries aren't charged, since it won't work properly.
- There is a bit of a burden to business owners to always remember to keep the batteries fresh so the system operates properly when you need it most.

11. CONCLUSION

Early fire detection is best achieved by the installation and maintenance of fire detection equipment in all areas of the forest.

12. FUTURE SCOPE

The future will be with multicriteria detection in which the detector will be more of a sensor, with the detection more for the products of combustion, such as carbon monoxide, carbon dioxide, sulfur dioxide, nitrogen oxides in addition to heat and particulate matter.

13. APPENDIX

Source Code

```
from google.colab import drive
drive.mount('/content/drive')
!pip install tensorflow
!pip install opencv-python
!pip install opencv-contrib-python
import tensorflow as tf
import numpy as np
from tensorflow import keras
import os
import cv2
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.preprocessing import image
train=ImageDataGenerator(rescale=1./255,
                                 shear_range=0.2,
                                 rotation_range=180,
                                 zoom_range=0.2,
                                 horizontal_flip=True)
train = ImageDataGenerator(rescale=1/255)
test = ImageDataGenerator(rescale=1/255)
train_dataset = train.flow_from_directory("/content/drive/MyDrive/Dataset/train
_set",
                                          target_size=(128,128),
                                          batch_size = 32,
                                          class_mode = 'binary' )
test_dataset = test.flow_from_directory("/content/drive/MyDrive/Dataset/test_se
t",
                                          target_size=(128,128),
                                          batch_size = 32,
                                          class_mode = 'binary' )
```

```
test_dataset.class_indices
#to define linear initialisation import sequential
from keras.models import Sequential
#to add layer import Dense
from keras.layers import Dense
#to create convolution kernel import convolution2D
from keras.layers import Convolution2D
#import Maxpooling layer
from keras.layers import MaxPooling2D
#import flatten layer
from keras.layers import Flatten
import warnings
warnings.filterwarnings('ignore')
model = keras.Sequential()
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Convolution2D(32,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Convolution2D(32,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Convolution2D(32,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.add(Dense(150,activation='relu'))

model.add(Dense(1,activation='sigmoid'))
model.compile(loss = 'binary_crossentropy',
              optimizer = "adam",
              metrics = ["accuracy"])
r = model.fit(train_dataset, epochs = 5, validation_data = test_dataset)
predictions = model.predict(test_dataset)
predictions = np.round(predictions)
predictions
print(len(predictions))
model.save("/content/forest1.h5")
#import load_model from keras.model
from keras.models import load_model
#import image class from keras
import tensorflow as tf
from tensorflow.keras.preprocessing import image
#import numpy
import numpy as np
#import cv2
import cv2
model = load_model("/content/forest1.h5")
import matplotlib.pyplot as plt
plt.plot(r.history['loss'],label='loss')
plt.plot(r.history['val_loss'],label='val_loss')
```

```
plt.legend()
plt.plot(r.history['accuracy'],label='acc')
plt.plot(r.history['val_accuracy'],label='val_acc')
plt.legend()
def predictImage(filename):
  img1=image.load_img(filename,target_size=(128,128))
  plt.imshow(img1)
  y=image.img_to_array(img1)
  x=np.expand_dims(y,axis=0)
  val=model.predict(x)
  print(val)
  if val==0:
    plt.xlabel(" NO fire",fontsize=30)
  elif val==1:
    plt.xlabel("fire",fontsize=30)
predictImage("/content/drive/MyDrive/Dataset/test_set/with fire/Forest_Fire (3)
.jpg")
predictImage("/content/drive/MyDrive/Dataset/test_set/forest/01_NeilBurnell_Mys
tical_photoverticall.jpg")
pip install twilio
pip install playsound
pip install opencv-python
#import opencv librariy
import cv2
#import numpy
import numpy as np
#import image function from keras
from keras.preprocessing import image
#import load_model from keras
from keras.models import load_model
#import client from twilio API
from twilio.rest import Client
#imort playsound package
from playsound import playsound
#load the saved model
model = load_model(r'/content/forest1.h5')
#define video
video = cv2.VideoCapture('/content/forest fire.mp4')
#define the features
name = ['forest','with forest']
video.isOpened()
from tensorflow.keras.preprocessing import image
from IPython.display import Audio
while(video.isOpened()):
  success,frame=video.read()
  cv2.imwrite("image.jpg",frame)
  img=image.load_img("/content/image.jpg",target_size=(128,128))
  x=image.img_to_array(img)
```

```
  x=np.expand_dims(x,axis=0)
  pred=model.predict(x)
  p=pred[0]
  print(pred)
  cv2.putText(frame,"predicted class = ",(100,100),cv2.FONT_HERSHEY_SIMPLEX, 1,
 (0,0,0), 1)
  if pred[0]==1:
    account_sid='AC1a92521871480f58548ab47433527298'
    auth_token='596097e117fb2295a39c05a192353001'
    client=Client(account_sid,auth_token)
    message=client.messages \
    .create(
        body="Forest fire is detected ,stay alert",
        from_='+14258421887',
        to='+919150947787')
    print(message.sid)
    print('Fire detected')
    print('SMS sent')
    wn=Audio('/content/tornado-siren.mp3',autoplay=True)
    display(wn)
    break
  else:
    print('No danger')
    break
  if cv2.waitKey(1) & 0xFF==ord('a'):
    break
video.release()
cv2.destroyAllWindows()
```

Github and Project Demo Link

GitHub Link

https://github.com/IBM-EPBL/IBM-Project-8655-1658926601

Demo Link:

https://www.loom.com/share/37c59fd8bb534743bb6fe32182ac1f92