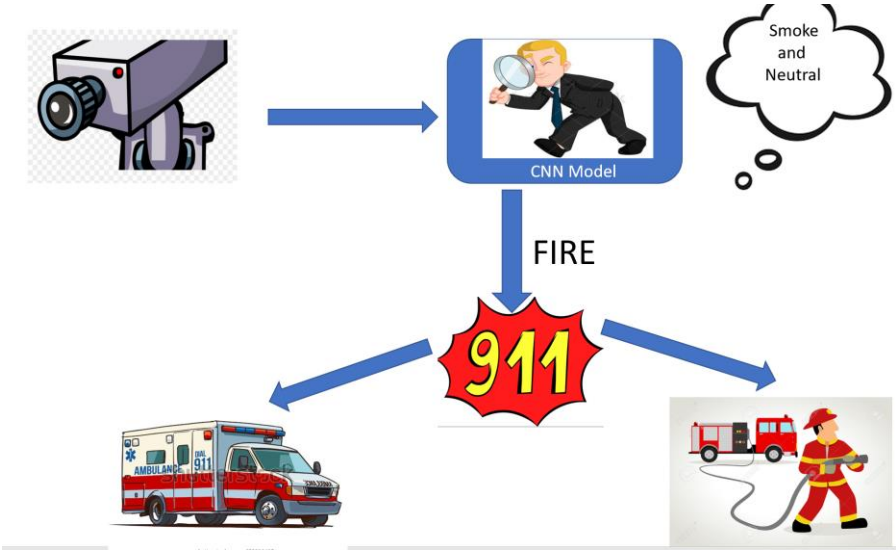
BLAZE SIGHT

(Let's Blaze the way to keep the blaze away)

CMPE 256 Summer 2020



Apoorv Mehrotra	**014597635**
Mohmmadsalman Mal	**014597609**
Pranjay Sagar	**014611922**
Sourabh Garg	**013727103**

Contents

Abstract

Fire safety has consistently been the piece of the most extreme significance in all regions over the globe. There are smoke alarms that have been embedded in all the potential spots to raise a crisis alert as ahead of schedule as could reasonably be expected. Nonetheless, there remains the extent of a downpour of events that are lethal and go unnoticed at the beginning phases. The result is shocking. Observing non-military personnel fire calamity announced by US Fire Association "1288 regular citizen home fire fatalities detailed by U.S. news media between January 1, 2020, and August 6, 2020". Withdrawing a year behind, in 2019, there was a huge fire misfortune in the Amazon woods. It wrecked 17 percent of the backwoods. This features the emergency that remaining parts undealt at places where alarms and smoke alarms can't be introduced at this point.

Thinking about every single such situation, it has gotten a matter of prime significance to guarantee fire safety. Be it a corporate working center point or private quarters spread across immense regions, fire safety setup is mandatorily deployed and managed. The current framework uses alarm hardware to raise a crisis and brisk call of activity. This has been a lot of help. Be that as it may, the current arrangement doesn't cover the open zones from forestalling any harm brought about by the blasting flares of fire.

There stands a decent extent of an upgrade in the field of the discovery of fire mostly in zones where alarm frameworks, smoke alarms couldn't be set up. This brought forth a lot of musings to conquer the unsolved issue. An inclining method that is being one of the territories of enthusiasm of the specialists and data scientists is the discovery and separation of fire situations through video content catch.

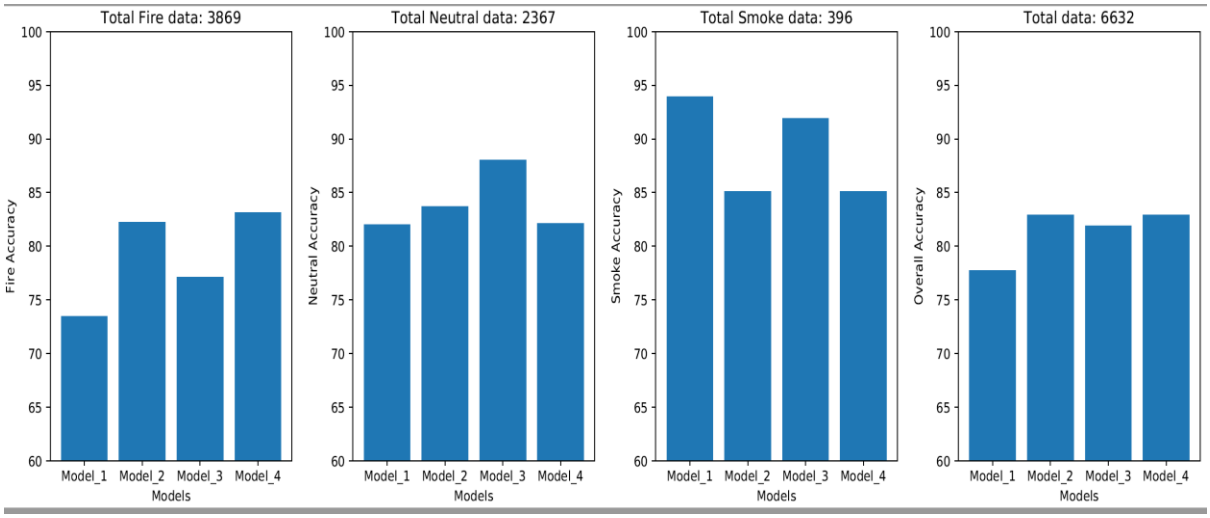
Numerous researchers have distributed their attempts to catch the video outlines and distinguish if there is a fire situation using profound learning models. A significant test in the zone is to recognize among fire, smoke, and impartial situations. The plan is to guarantee the assurance from fire.

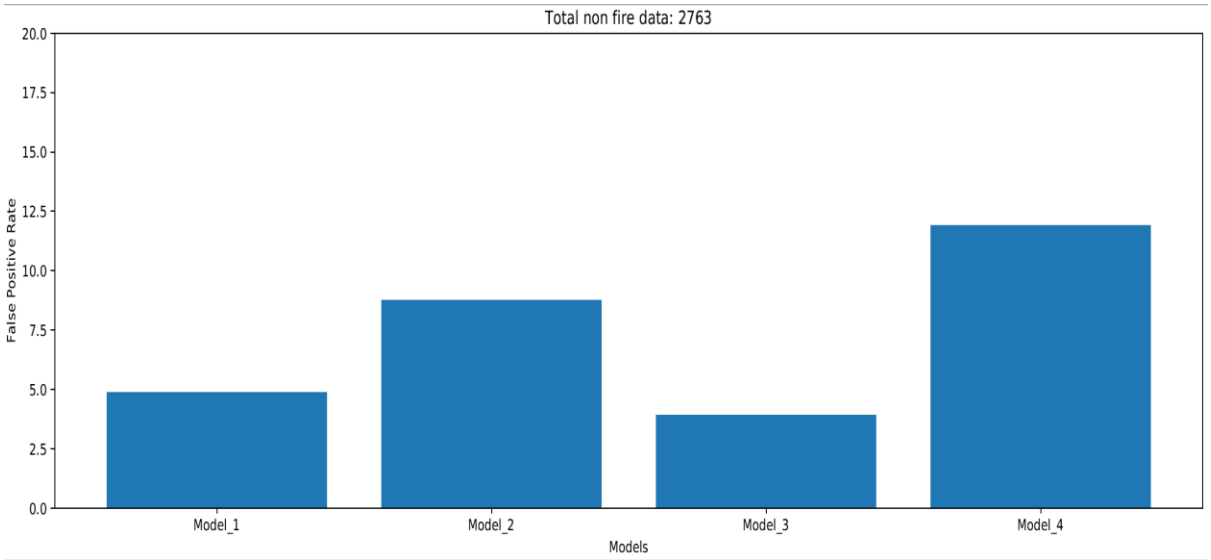
This project manages the execution of combined rationale from IEEE papers and self-decisions to build up a framework that catches recordings at the front end. It at that point breaks it into the little edges and passes it to the CNN model to distinguish the criticality of the circumstance. On the off chance that fire is seen in different sequential casings, it reports to the crisis administrations with a quick impact to battle the circumstance and re-establish regularity.

Contribution Distribution

Apoorv Mehrotra (014597635) (ML model and front-end sub team member)

1. Contributed the dataset creation. It involved the segregation of scenarios between fire, neutral, and smoke.
2. Worked on using Transfer learning and created multiple models with a varied hyper-parameters using vgg16 as the base model. Out of them selected the second model that had the best performance.

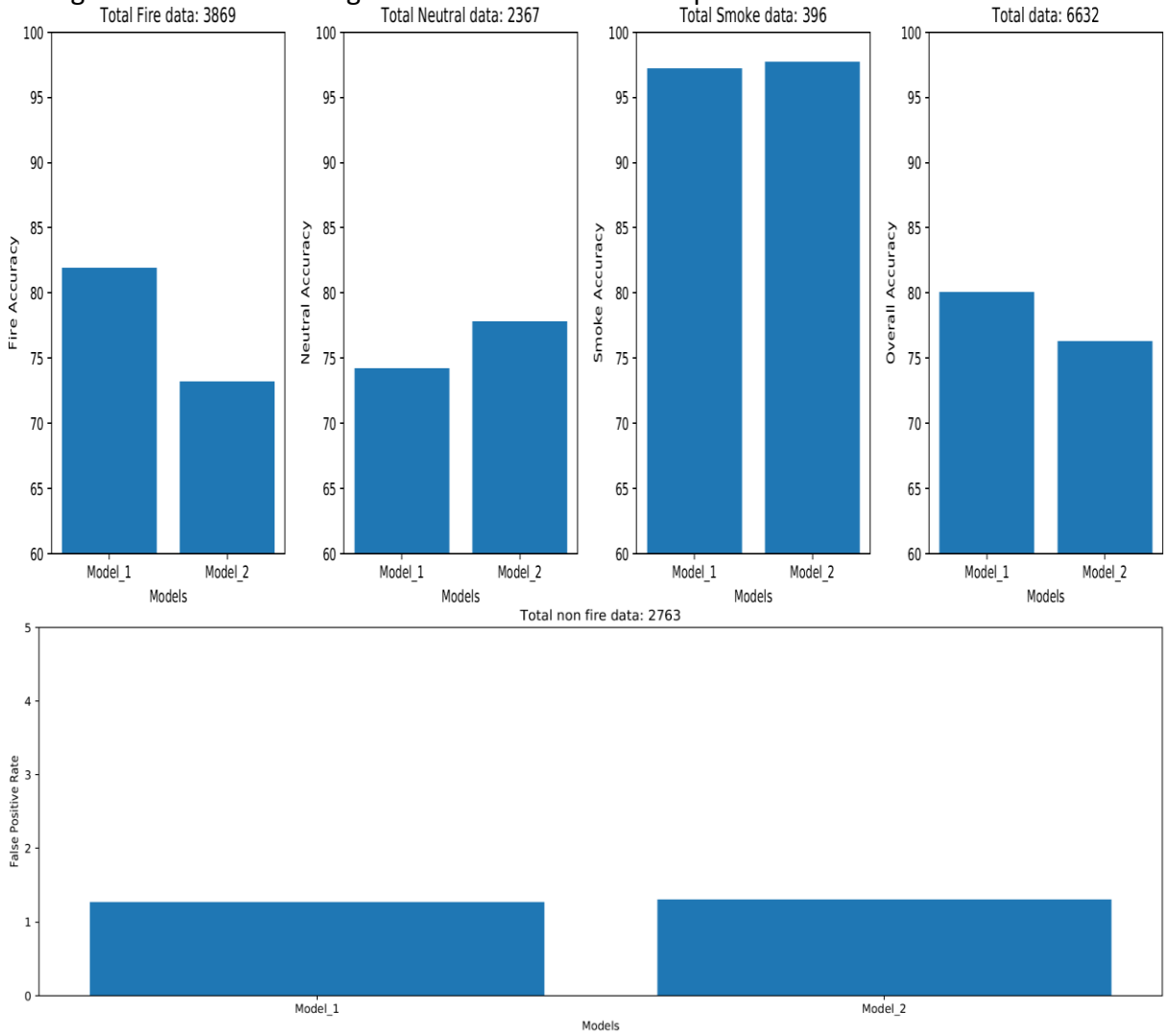




3. Worked in collaboration with Mohmmandsalman Mal to develop the front end using the React framework. As an individual responsibility handled the local storage of frame detection result and calculation based on the threshold. As a sub-team for the front end, handled video upload, live feeds through the camera and making the prediction. If the Prediction was fire, calling the alert APIs handled by the back-end sub-team.

Mohmmadsalman Mal (014597609) (ML model and front-end sub team member)

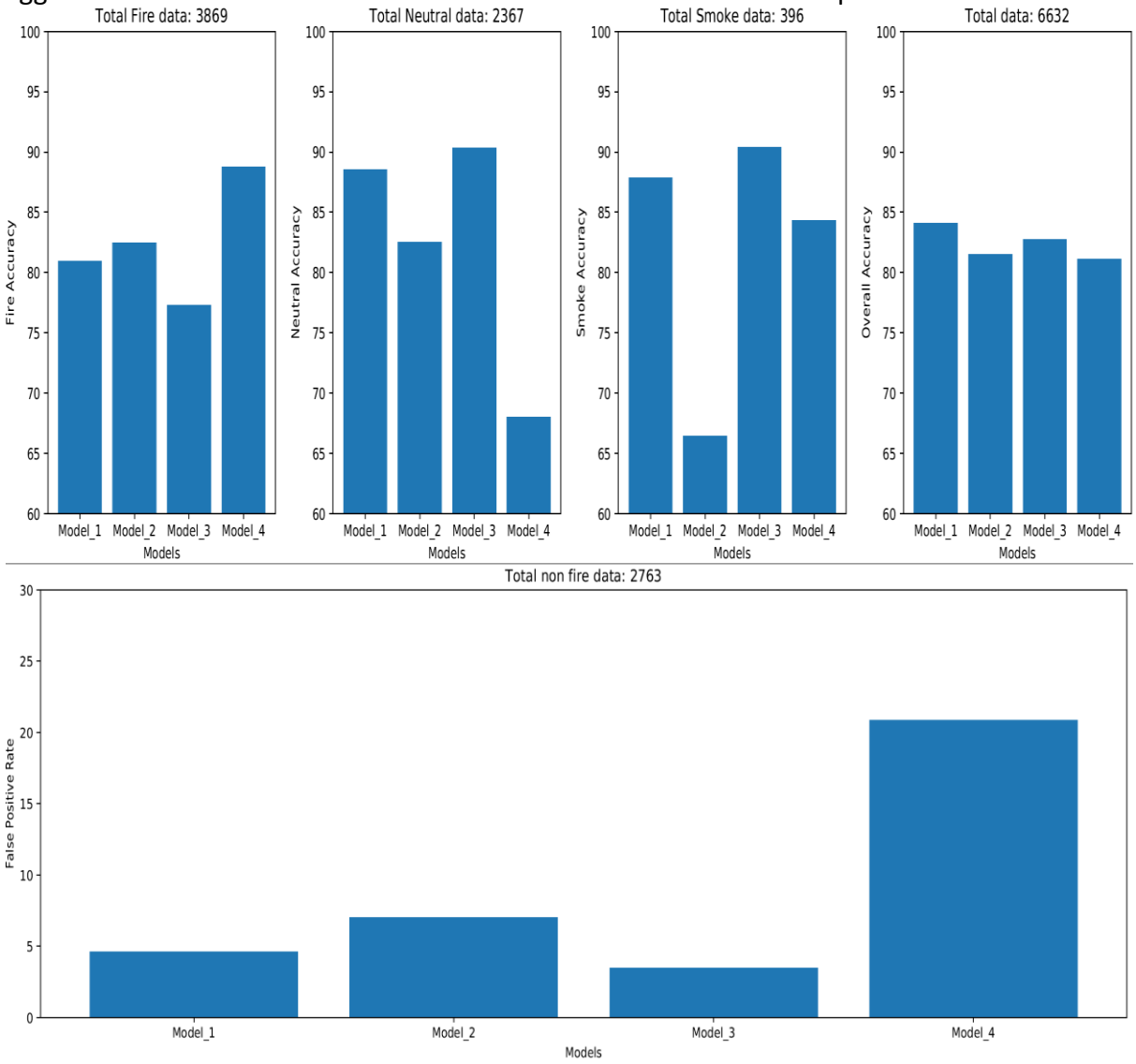
1. Worked in the dataset collection. It comprised of filtering the captured images between the three defined classes.
2. Created multiple models referring to a sample model. Chose the best out of the configured ones considering the one that had the best performance.



3. Worked in collaboration with Apoorv Mehrotra to develop the front end using the React framework. As an individual responsibility handled the creation of frames from the video captured to be sent to the blaze sight model for analysis. As a sub-team for the front end, handled video upload, live feeds through the camera and making the prediction. If the Prediction was fire, calling the alert APIs handled by the back-end sub-team.

Pranjay Sagar (014611922) (ML model and back-end sub team member)

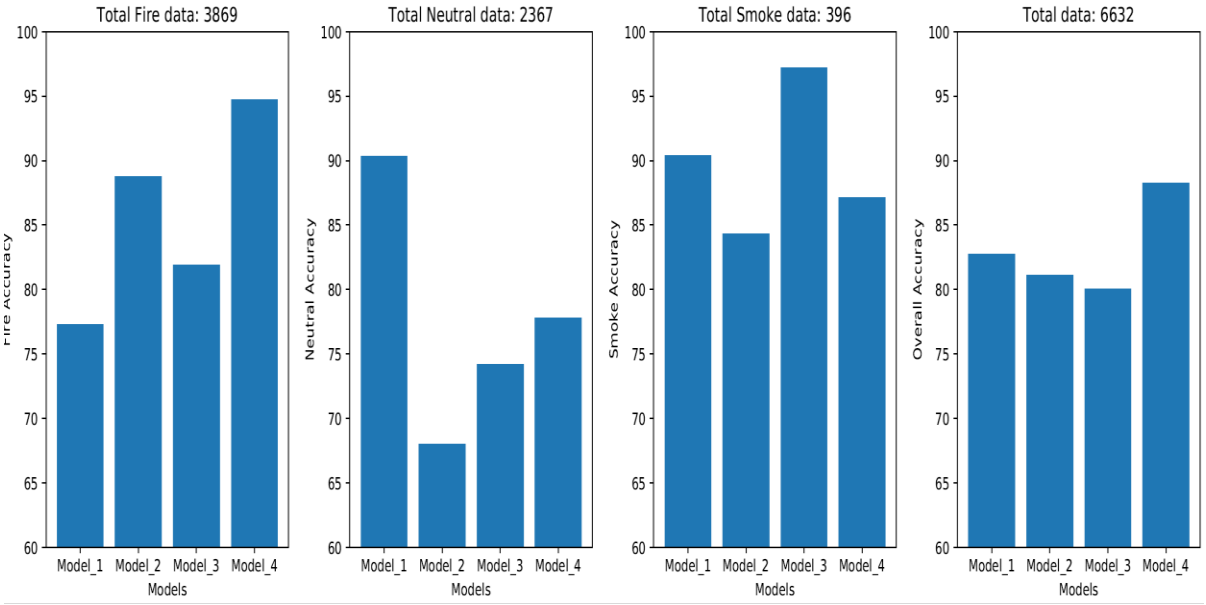
1. Dataset creation was the first task taken care of. It implied putting the correct frames in the specified classes.
2. Designed 4 machine learning models that were based on Transfer Learning. Utilized vgg19 as the base model. Selected the one that had the best overall performance.

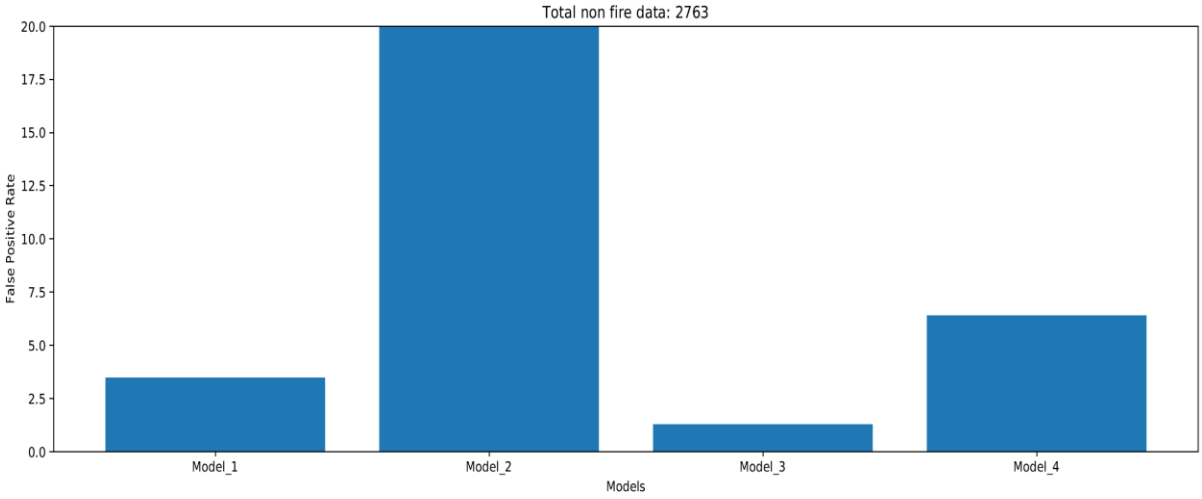


3. As a part of the back-end team, had worked in collaboration with Sourabh Garg, to handle the calls in back-end send through APIs and model loading. As an individual responsibility handled the receiving of multiple frames captured from the video. Proper management of sequential prediction of the frames received from among the defined classes (fire, smoke, neutral) and sending it back in the same queue.

Sourabh Garg (013727103) (ML model and back-end sub team member)

1. Concentrated on defining the correct images and video stills from various sources to the appropriate classes.
2. Re-layered the existing sample model. Selected the best among all counting the factors of overall efficiency. Used different hyper-parameters to train different models.



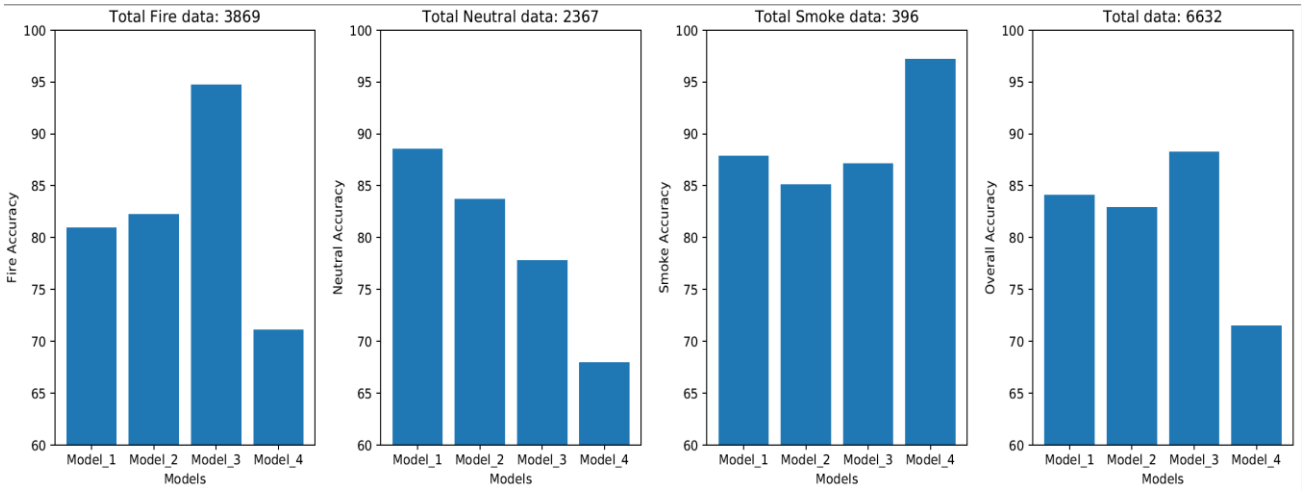


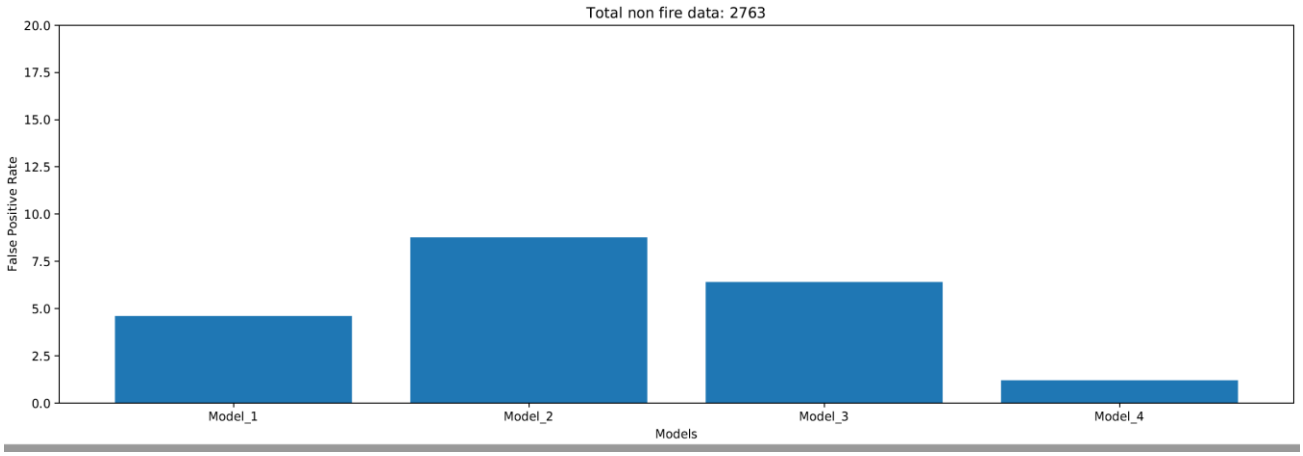
3. As a part of the back-end team, had worked in collaboration with Pranjay Sagar, to handle the calls to back end through APIs and model loading. As an individual responsibility handled the alert call handling in case of fire detection. Whenever a fire detection response is received by front-end after prediction, it triggers a call to alert API, handled the generation of alert text and emails to 911 for immediate attention.

Combined Team Task

1. Designing the workflow for the project and brainstorming on various approaches to formulate the complete pipeline.

2. Loading the best model of all the team members saved in the repository and judging it on various parameters as mentioned below to select the appropriate model according to the use case.
    a. Fire Detection Accuracy
    b. Neutral Detection Accuracy
    c. Smoke Detection Accuracy
    d. Overall Accuracy
    e. False Positives generated

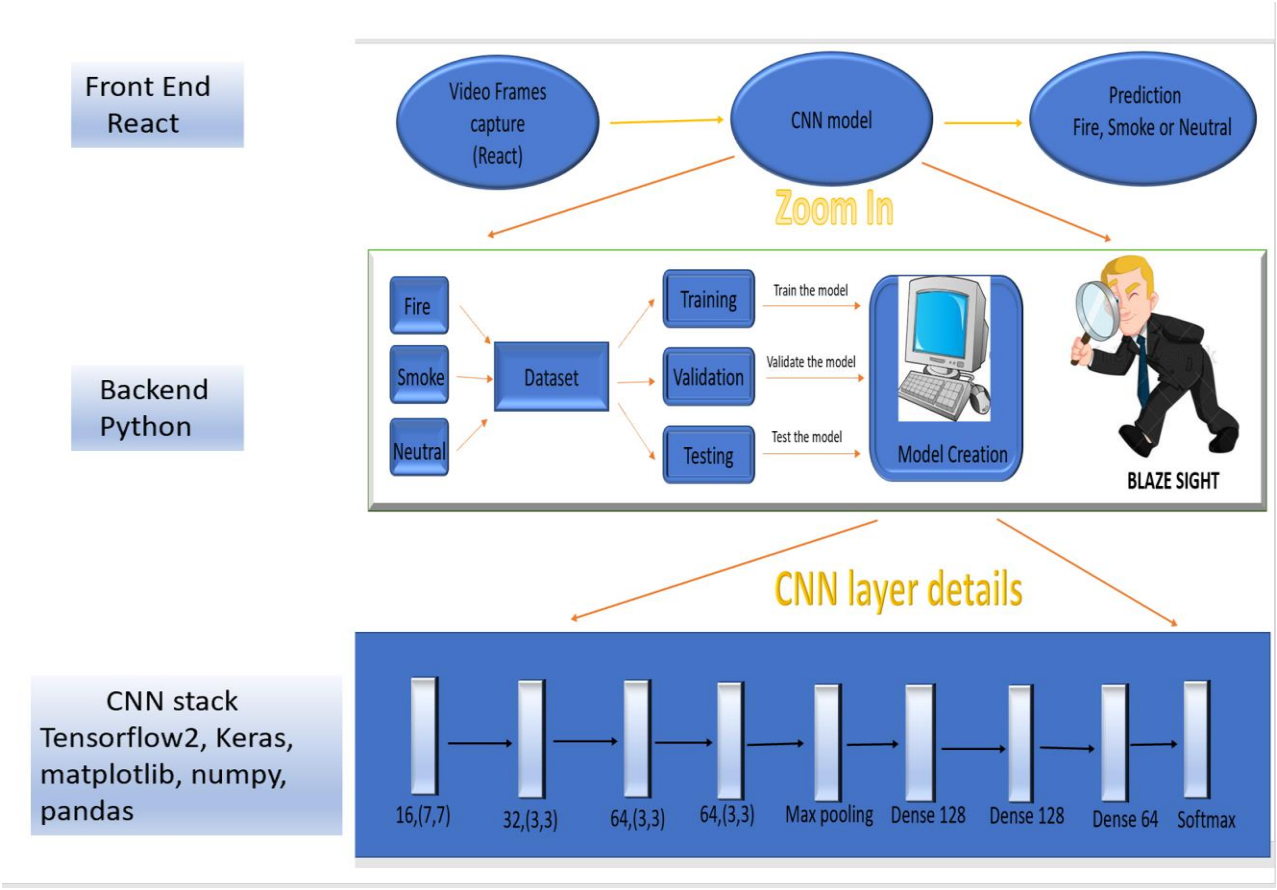




3. Report content generation for the project depicting the pathway to attain the best use of the designed project.

Architecture Diagram

Below is the hierarchical architectural diagram highlighting the detailed workflow and Technology Stack.



Model Creation

Referring to IEEE papers to understand the concept of the Deep CNN model planted the roots for the development and creation of the Blaze Sight model. After running through multiple model creation, the best one chosen had the below configuration.

Libraries used: tensorflow, keras, matplotlib, numpy, pandas

Platform used for Training: HPC

Layer Details as under

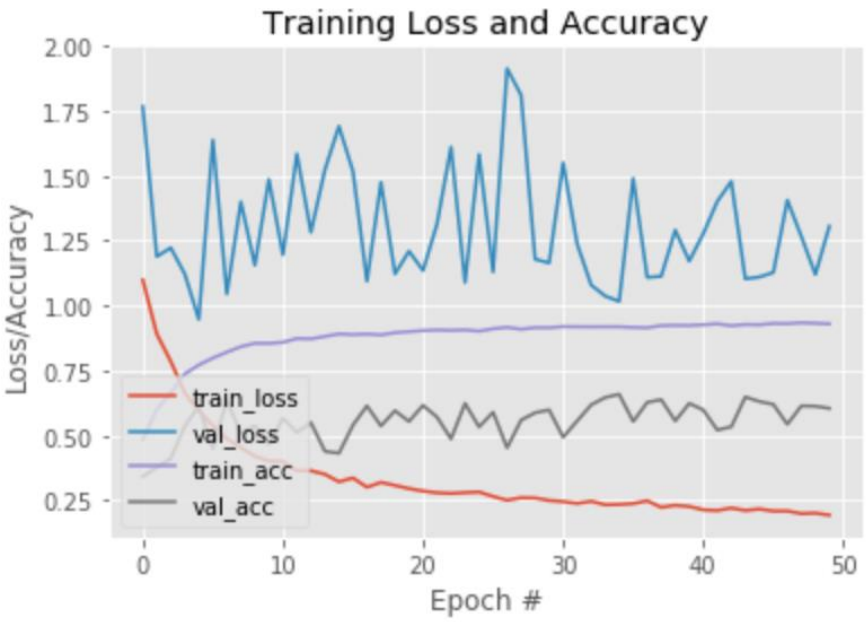
```
def build_model0():
  model = models.Sequential()
  changeDim = -1
  #Layer1
  model.add(layers.SeparableConv2D(16, (7, 7), activation='relu', padding='same', input_shape=(IMAGE_SIZE, IMAGE_SIZE, 3)))
  model.add(layers.BatchNormalization(axis=changeDim))
  model.add(layers.MaxPooling2D(2,2))

  #Layer2
  model.add(layers.SeparableConv2D(32, (3, 3), activation='relu', padding='same'))
  model.add(layers.BatchNormalization(axis=changeDim))
  model.add(layers.MaxPooling2D(2,2))

  #Layer3
  model.add(layers.SeparableConv2D(64, (3, 3), activation='relu', padding='same'))
  model.add(layers.BatchNormalization(axis=changeDim))

  #Layer4
  model.add(layers.SeparableConv2D(64, (3, 3), activation='relu', padding='same'))
  model.add(layers.BatchNormalization(axis=changeDim))

  #Layer5 MaxPooling
  model.add(layers.MaxPooling2D(2,2))

  #First Fully Connected Layer
  model.add(layers.Flatten())
  model.add(layers.Dense(128, activation='relu'))
  model.add(layers.BatchNormalization())
  model.add(layers.Dropout(0.5))

  model.add(layers.Dense(128, activation='relu'))
  model.add(layers.BatchNormalization())
  model.add(layers.Dropout(0.5))

  model.add(layers.Dense(64, activation='relu'))
  model.add(layers.BatchNormalization())
  model.add(layers.Dropout(0.5))

  model.add(layers.Dense(numclasses, activation='softmax'))
```

Categorical cross-entropy has been used to tabulate the losses. It suggests one hot encoded practice, indicating the final classification as a single class.

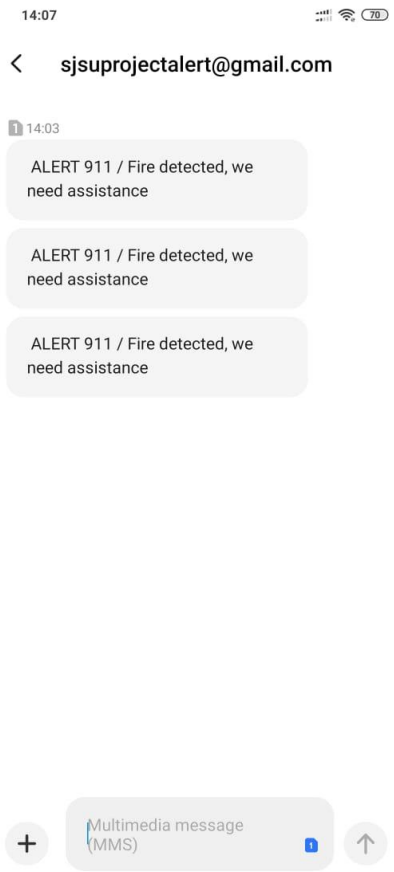
Stochastic Gradient Descent (SGD) is used as an optimizer with *0.015* learning rate and a momentum of *0.9*. Momentum prevents the settling down at saddle point and continues to locate the global minima. A decay rate of *0.015/50* preventing the weights from growing too large.

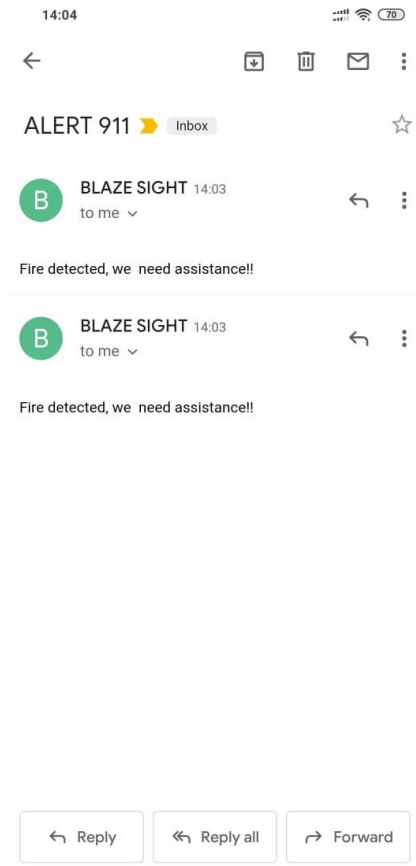
```
  loss = losses.categorical_crossentropy#Losses.sparse_categorical_crossentropy
  #one-hot encoded, use categorical_crossentropy. Examples (for a 3-class classification): [1,0,0] , [0,1,0], [0,0,1] But if your Yi's ar
e integers, use sparse_categorical_crossentropy. Examples for above 3-class classification problem: [1] , [2], [3]
  optimizer = optimizers.SGD(lr=0.015, momentum=0.9,decay=0.015/50)#optimizers.RMSprop(lr=1e-4)#optimizers.SGD(lr=0.001) #Adam(lr=0.001)
  model.compile(loss=loss,
              optimizer=optimizer,
              metrics=['acc'])
```

The graph showcasing the training and validation accuracy is as under



As a part of the next phase implementation, whenever fire detection takes place, it generates an alert. This causes the delivery of information to 911 through SMS and email alerts.





Challenges

There have been major achievements in this project in a very short period. However, reaching up to the current accuracy was a path full of challenges. Some of which are illustrated as under:

1. Detailing, isolation, determination of information from numerous assets to shape the dataset was one of the greatest and most tedious exercises. There were various assets investigated to get the important conditions of the classes characterized. They were as yet made out of recordings to get the opportunity to acquire the photos for the model readiness. Among the characterized class, information for smoke identification and separation from the fire was the undertaking which required the most extreme consideration.
2. To acquire great precision in identification, there were numerous model choices picked. An intensive perusing and usage of IEEE papers. It was trailed by executing Transfer Learning models that included vgg16, vgg19. Characterizing the new models was additionally an answer thoroughly considered to get the most ideal precision.
3. Settling on the hyper-parameters to get the best preparing system for the model devoured a lot of time and brainstorming.
4. Another test that attracted the consideration was to choose the breaking of video into outlines at a normal period and investigate numerous casings before settling on the last strategy.

Future Enhancements

As a future enhancement for this project, the below recommendations can be considered.

1. The efficiency of detection could be slightly improved by incorporating different values of the filters.
2. Utilizing the concept of edge computing, the exact location can also be tracked and reported with an immediate effect.
3. Sub-categorise the fire class on the basis of its intensity, so ensure the alerting about the fire that is drastic and damage-causing.

Conclusion

Detecting fire from live video catching could be of incredible assistance and rescuer of life and assets all over the world. The fundamental advantage it gives over the typical smoke alarms is that it could be introduced in satellites and lamp posts to cover the whole open territories, including grounds, woods.

References

1. Xiong Hao, Cong Linhu, He Weixin, Video Flame Detection Using Convolutional Neural Networks, 2019 IEEE 4th International Conference on Image, Vision and Computing Naval Aviation University, Yantai, China
2. Khan Muhammad, Jamil Ahmad, Irfan Mehmood, Seungmin Rho, and Sung Wook Baik, "Convolutional Neural Networks Based Fire Detection in Surveillance Videos", Received January 30, 2018, accepted March 3, 2018, date of publication March 6, 2018, date of current version April 23, 2018
3. Kanwal K, Liaquat A, Mughal M, et al. Towards Development of a Low Cost Early Fire Detection System Using Wireless Sensor Network and Machine Vision[J]. Çetin A E, Dimitropoulos K, Gouverneur B, et al
4. Video fire detection – Review[J]. Digital Signal Processing, 2013, 23(6):1827-1843
5. https://www.kaggle.com/phylake1337/fire-dataset
6. https://github.com/cair/Fire-Detection-Image-Dataset
7. https://collections.durham.ac.uk/files/r2d217qp536#.XwEUc55KhQI
8. http://smoke.ustc.edu.cn/datasets.htm
9. https://www.thefreshquotes.com/fire-safety-slogans/#:~:text=%E2%80%9CFire%20Destruction%20Is%20One%20Man's,Be%20Alert%2C%20Avert%20Fire.%E2%80%9D&text=%E2%80%9CLet's%20Blaze%20The%20Way%2C%20To,But%20Always%20See%20It%20Out.%E2%80%9D
10. https://www.pyimagesearch.com/2019/11/18/fire-and-smoke-detection-with-keras-and-deep-learning/
11. https://www.pinclipart.com/pindetail/bTRwJb_important-features-for-security-camera-transparent-security-camera/
12. https://friendlystock.com/product-tag/lens/
13. https://www.123rf.com/photo_34886185_stock-vector-cartoon-illustration-of-a-firefighter.html
14. https://www.123rf.com/photo_34886185_stock-vector-cartoon-illustration-of-a-firefighter.html
15. https://www.vectorstock.com/royalty-free-vector/emergency-911-icon-icon-cartoon-vector-13601127
16. https://www.vectorstock.com/royalty-free-vector/cartoon-desktop-computer-vector-21585135