Detecting Forest Fires using Machine Learning models and conducting predictive analysis employing Mathematical models.



By

**Azra Nisar**

In partial fulfilment of the requirement for the degree

Bachelor of Science in Computer Science

Department of Computer Science, School of Arts & Science

University of Central Asia, Naryn campus, Kyrgyz republic

(2024)

**DECLARATION BY AUTHOR**

I/we certify that this work has not been accepted in substance for any degree and is not concurrently being submitted for any degree other than that of Bachelor of Science in Computer Science being studied at the Department of Computer Science, School of Arts & Science, University of Central Asia, Kyrgyz Republic. I/we also declare that this work is the result of my/our own findings and investigations except where otherwise identified by references and that I/we have not plagiarized another’s work.

A close-up of a signature

Description automatically generated

Azra Nisar

**DECLARATION BY SUPERVISOR**

I, the undersigned hereby certify that I have read this project report and finally approve it with

recommendation that this report may be submitted by the authors above to the final year project

evaluation committee for final evaluation and presentation, in partial fulfillment of the

requirements for the degree of Bachelor of Science in Computer Science at the Department of

Computer Science, School of Arts & Sciences, University of Central Asia, Kyrgyz Republic.



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dr. Muhammad Fayaz

**Table of Contents**

[ABSTRACT 5](#_Toc163548312)

[Acknowledgment 6](#_Toc163548313)

[CHAPTER 1. INTRODUCTION 8](#_Toc163548314)

[1.1 Aims and Objectives 8](#_Toc163548315)

[1.2. Business benefits of the project 9](#_Toc163548316)

[CHAPTER 2. LITERATURE REVIEW 12](#_Toc163548317)

[CHAPTER 3. PROPOSED MODEL 14](#_Toc163548318)

[3.1 Data Acquisition, and Preprocessing 15](#_Toc163548319)

[3.1.1 Model selection and training 16](#_Toc163548320)

[3.1.2 Standardization 17](#_Toc163548321)

[3.2 Prediction 17](#_Toc163548322)

[3.2.1 Binary Convolution Neural Network 17](#_Toc163548323)

[3.2.2 Random Forest Tree 19](#_Toc163548324)

[3.2.3 Support Vector Machine 19](#_Toc163548325)

[3.2.4 ARIMA (MATHEMATICAL MODEL) 20](#_Toc163548326)

[3.5 Application Module 20](#_Toc163548327)

[CHAPTER 4. IMPLEMENTATION, RESULTS, AND DISCUSSION 21](#_Toc163548328)

[4.1 Implementation Setup 21](#_Toc163548329)

[4.1. Dataset 22](#_Toc163548330)

[4.2. Results 23](#_Toc163548331)

[4.2.1. Random Forest Algorithm 23](#_Toc163548332)

[4.2.2 Support Vector Machine 24](#_Toc163548333)

[4.2.3. Binary Convolution Neural Network 25](#_Toc163548334)

[CHAPTER 5. PERFORMANCE EVALUATION 25](#_Toc163548335)

[CHAPTER 6. APPLICATION MODULE 27](#_Toc163548336)

[6.1 Similar applications comparison table 30](#_Toc163548337)

[6.1.1 Links to applications 31](#_Toc163548338)

[6.2 Unique Features of my app 32](#_Toc163548339)

[6.3 Technical specifications of the project 32](#_Toc163548340)

[6.4 Project plan and schedule 33](#_Toc163548341)

[6.4.1 Work Breakdown of project 35](#_Toc163548342)

[6.4.2 Gannt Chart 36](#_Toc163548343)

[CHAPTER 7. RISK MANAGEMENT AND PLAN 37](#_Toc163548344)

[7.1 Four main risks 37](#_Toc163548345)

[7.1.1 The quality and quantity of data (Balance Data) 37](#_Toc163548346)

[7.1.2 Model Sturdiness and adaptability. 37](#_Toc163548347)

[7.1.3 Model Fairness and Biasness 38](#_Toc163548348)

[7.1.4 Equipment Malfunctioning 38](#_Toc163548349)

[CHAPTER 8. DATABASE DIAGRAMS 38](#_Toc163548350)

[References 41](#_Toc163548351)

**List of Figures**

[Figure 2. 1 A Conceptual Model of proposed Method 13](#_Toc163548352)

[Figure 3. 1 Detailed Processing Diagram of the Proposed Model 14](#_Toc163548469)

[Figure 3. 2 Detailed diagram for data preprocessing 15](#_Toc163548470)

[Figure 3. 3 Schematic diagram for CNN 18](#_Toc163548471)

[Figure 3. 4 Detailed diagram of Random forest tree 19](#_Toc163548472)

[Figure 3. 5 Schematic Diagram for SVM 20](#_Toc163548473)

[Figure 3. 8 Illustration of web-application for prediction 21](#_Toc163548474)

[Figure 4. 1 Fire and No Fire prediction confusion matrix using Random Forest. 23](#_Toc163548477)

[Figure 4. 2 Performance evaluation of Random Forest. 24](#_Toc163548478)

[Figure 4. 7 Confusion matrix for SVM. 24](#_Toc163548479)

[Figure 4. 8 Performance evaluation of SVM. 25](#_Toc163548480)

[Figure 6. 1 Initial home page of the interface 27](#_Toc163548504)

[Figure 6. 2 Dashboard page 28](#_Toc163548505)

[Figure 6. 3 Results display 28](#_Toc163548506)

**List of Tables**

[Table 1. 1 Notations with Descriptions 10](#_Toc163548420)

[Table 4. 1 Hardware and Software Specifications used in the proposed model implementations 21](#_Toc163548429)

[Table 5. 1 Performance evaluation of random forest algorithm 26](#_Toc163548451)

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# ABSTRACT

The escalating incidence of forest fires poses a significant threat to ecosystems and contributes to the cycle of global warming. This project delves into the intricate relationship between forest fires and global warming, exploring how rising temperatures facilitate the spread of fires while the ensuing heat exacerbates climate change, creating a positive feedback loop. Human-induced fires, which spread twice as fast as natural fires and result in greater arboreal mortality, have exacerbated this issue, as evidenced by research from the University of Maryland highlighting a loss of three million more hectares of forest since 2001—an area equivalent to Belgium (Alkhatib, 2023). Furthermore, the environmental ramifications of forest fires extend beyond deforestation; the smoke emitted introduces noxious pollutants such as ozone, NO2, PM2.5, and hydrocarbons into the atmosphere, significantly contributing to air pollution and global warming.

This research seeks to offer a comprehensive analysis of forest fire patterns and the detection of their occurrences through advanced machine learning techniques. By predicting forest fires with dynamic mathematical models, the project aims to equip authorities and stakeholders with tools for better forest management and mitigation strategies. In addressing this critical issue, the project stands to benefit not just specific regions, but global communities, underscoring the universal impact of wildfires and the collective responsibility in combating them.

Richard C. Rothermel introduced his groundbreaking fire spread rate model in 1972 which is recognized as the universal model to predict wildfire behavior (firelab, 2023). Rothermel's model was developed during the 1960s and 1970s at the Missoula Fire Sciences Laboratory. Known as the "Rothermel Model," it forms the basis of numerous digital fire behavior analysis systems, which are integral to fire management, training, and operational forecasting. In the initial stages of its development, the groundbreaking Rothermel Model was seen as having a broad spectrum of potential applications. These included predicting fire behavior on a localized scale, simulating wildfires on a large scale, and strategizing for fuel treatments and controlled burns (firelab, 2023). Over the following decades, these envisioned uses, among others, have been realized and integrated into practical operations.

***Keywords: Machine learning, forest fire, user interface, dataset, mathematical models, ARIMA, Rothermel model***

# Acknowledgment

Throughout the final year project, I have received huge support and assistance. I would first like to thank my supervisor, Dr. Muhammad Fayaz, whose expertise were invaluable from the step of formulating the research topic to the end of this great work. Your insightful and experienced feedback and support pushed me to sharpen my understanding as well as the perfect culmination of this research.

Other than that, I would also want to extend my utmost gratitude to all the Computer Science Department staff, especially our department head Dr. Ayman Aljarbouh for helping us in every possible way to complete our work without any administrative hurdles.

Lastly, I would also like to take this opportunity to express my gratitude and love for my parents for their unwavering support during all the challenges I faced relevant to the given thesis.

# 

# 

# CHAPTER 1. INTRODUCTION

Forest fire is a burning fire in an area covered with trees which is challenging to control and spreads at a fast pace (Cambridge reference, n.d.). Global warming and forest fire are profoundly interconnected. For instance, global warming results in forest fire due to the rising temperature and the heat of forest fire eventually contributes to global warming. Therefore, their relationship forms a positive feedback loop. Apart from global warming, human activities are equally responsible for forest fire. According to University of California’s fire ecologist fires sparked due to human activities are more extreme because they spread faster and kills a greater number of trees than the ones caused by lightning or nature (Joosse, 2020). For instance, human prone fires spread at a speed of 1.8km per day which is twice as fast as lightning induced fires. The number of forest fire has increased drastically in the past few years and its impacts have amused everyone indeed. According to a research study conducted by the university of Maryland, forest fires have resulted in a loss of three million more hectares of trees compared to 2001 which equals an area of roughly Belgium’s size (MacCarthy et al., 2023). It is not only the fire that leads to global problems, but wildfire smoke also has hazardous impacts on the environment. The smoke contains harmful air pollutants such as ozone, NO2, PM2.5, and hydrocarbons which results in air pollution eventually (WHO, n.d.). The carbon released from wildfires is a huge contributor to global warming and climate change.

Wildfires affect everyone equally regardless of where we live therefore, my research and project will address this respective global issue. The research will help in analyzing and finding trends in forest fires and detecting their occurrences using various machine learning techniques. Furthermore, this project aims to predict forest fires using dynamic mathematical models. I have used Rothermel model to find the spread rate of the forest fire and ARIMA (Autoregressive integrated moving average) model to conduct predictive analysis.

## Aims and Objectives

* **Aims:**
  + This project aims at training a machine learning model on an image dataset collected from an online source which can recognize the fire in an image and developing an alert system to speed up the process of fire extinguishing, which can be enhanced and monetized and used by the fire extinguishing companies, by generating an alert that notifies the fire extinguishing companies.
* **Objectives:**
* Create an alert system in the form of web application or mobile app to notify the relevant personals about fire.
* Train a deep learning model for fire recognition to increase accuracy and decrease technical risks of misclassification.
* This project aims to acquire a minimum of 80% accuracy for the classification model.
* To employ classification models to classify fire, no fire, smoke, and no smoke images from the dataset.
* To use dynamic mathematical models to predict forest fires.

## 

## 1.2. Business benefits of the project

The business benefits of the project focused on analyzing and predicting forest fires using machine learning and dynamic mathematical models are multifaceted, extending from economic gains to corporate social responsibility:

* *Risk management and management.*

Businesses in sectors like insurance, agriculture, and real estate can significantly reduce risks and financial losses associated with forest fires by implementing predictive models that allow for better planning and quicker response.

* *Resource Allocation*

Efficiently predicting forest fires enables governments and firefighting units to optimally allocate resources, thus saving costs associated with emergency responses and reducing the economic impact of fires.

* *Benefits for the Insurance Industry*

For insurance companies, improved prediction models mean more accurate risk assessment, which can lead to more tailored insurance packages and premiums, potentially increasing profitability.

* *Real Estate and land value protection*

Real estate developers and investors can use insights from this project to protect their assets and invest in safer areas, preserving property values and investor confidence.

* *Tourism and recreation*

For businesses in the tourism sector, better management of forest fire risks can protect natural attractions and ensure the continuity of tourism revenues.

* *Regulatory Compliance*

For businesses with operations in fire-prone areas, advanced prediction capabilities can help in maintaining compliance with environmental regulations and avoiding fines or sanctions.

* *New markets and innovation*

Technology firms can utilize the findings to develop innovative products, such as early warning systems and firefighting drones, opening new markets and revenue streams.

Table 1. 1 Notations with Descriptions

|  |  |
| --- | --- |
| **Notations** | **Explanations** |
| ROS | Rate of Spread |
| WHO | World Health organization |
| ML | Machine Learning |
| PM | Particulate Matters |
| RF | Random Forest |
| SVM | Support Vector Machine |
| CNN | Convolutional Neural Networks |
| ARIMA | Autoregressive Integrated moving average |
| SARIMA | Seasonal Autoregressive Integrated moving average |
|  |  |

The structure of the remaining report is given below. In Section 2 the literature review of different models for energy consumption prediction in smart homes has been done in detail. Section 3 presents the proposed model, the sub-modules, such as preprocessing, prediction, performance evaluation, interface have also been discussed in detail in this section. In Section 4 the implementation details and results are given. Section 5 presents the performance evaluation, and Section 6 represents the application module. In the last Section 7 the proposed work is concluded.

# CHAPTER 2. LITERATURE REVIEW

Forests cover a wide range of area on earth, and with an increase in global warming and climate change forests conservation has become a major concern. Unfortunately, due to natural causes and human activities forest fires are increasing at a fast pace thus affecting the whole ecosystem. Uncontrollable flames caused an estimated 4,225,000 km2 of land to burn between 2002 and 2016 (Abdusalomov et al., 2023). Some of the main natural causes of forest fires are lightning, volcanos, dry weather, and wind while smoking and cooking are human activities that contribute to wildfires. Amusingly 90% of the forest fires are caused by human activities (Abdusalomov et al., 2023).

Machine learning techniques have been widely used in forest fire prediction and detection. Fire detection systems based on machine learning rely on manually extracting visible information from photos. These features solely focus on the superficial aspects of the flame, which may result in data loss when manually extracted. Deep learning algorithms, unlike machine learning algorithms, can automatically extract and learn complex feature representations (Sathishkumar et al., 2023).

CNNs have significantly improved performance in a variety of computer-based vision applications, including visual identification and image categorization. Many researchers have employed CNN (convolution neural network) based wildfires detection using unmanned aerial vehicle dataset (Sathishkumar et al., 2023). Convolutional Neural Networks (CNN) are a sort of Artificial Intelligence (AI) technique that has been demonstrated to perform better than cutting-edge approaches in image classification and other computer vision applications, but their training time may be prohibitive, furthermore, when there is an insufficient dataset, a pre-trained CNN may underperform (Seydi et al., 2022).

Most of the fire detection research have used YOLOv2 (you only look once) CNN for both outdoor and indoor fire scenarios. (YOLO) is a deep learning model for object detection; YOLOv2 is the next version that has been upgraded to address YOLO's shortcomings, namely the inability to accurately locate and mark the region of interest in images and the lower recall rate when compared to other region-oriented algorithms. Increasing the architecture's effectiveness as a result.

According to a research study, automatic fire detection can be classified into three types: aerial, ground, and borne detection. The ground-based systems use numerous gazing black and white video cameras in fire detection, which detect smoke and compare it to natural smoke (Rajan et al., 2022). The key advantage of employing this technology is the great temporal and spatial resolution, thus, making it easier to detect. However, these techniques still come with some demerits therefore, it is vital to use a mechanism which can detect the fire as early as possible.

Mathematical models have been useful to predict the spread rate of fire. There are mainly three types of mathematical models which include empirical, semi empirical, and theoretical models. Theoretical approaches to understanding fire dynamics involve applying the principles of combustion, fluid dynamics, and heat transfer. These approaches necessitate incorporating numerous parameters into the computational equations. The resulting equations are typically complex, complicating their practical application, particularly in real-world scenarios in developing nations. Furthermore, confirming the accuracy of these models is challenging, given the diverse and large-scale nature of wildland fires (firelab, 2023). Empirical models are built upon statistical correlations derived from observed data and past studies on wildland fires. Their application is generally confined to scenarios that closely resemble the tested conditions. Meanwhile, semi-empirical models combine basic physical principles with data obtained from experiments to enhance their applicability (firelab, 2023).

The Rothermel Model, renowned for its effectiveness both in practical scenarios and theoretical applications, remains a cornerstone in contemporary fire behavior modeling. Its integration into various advanced modeling techniques, including remote sensing, finite difference methods, finite element analysis, neural networks, and cellular automata, highlights its fundamental role. However, the model's complexity, characterized by the need for 24 distinct parameters, renders its implementation quite costly.

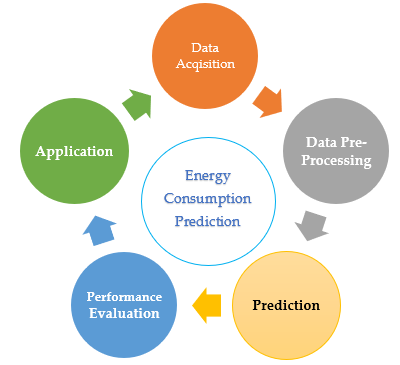


Figure 2. A Conceptual Model of proposed Method

# CHAPTER 3. PROPOSED MODEL

* 1. *Iterative Method*

Flexibility in Project Evolution: The iterative approach enables you to seamlessly integrate new changes and insights at different stages of the project. This is especially valuable for a project that combines image datasets with predictive modeling due to its inherent complexity.

Progressive Testing and Enhancement: In this model, you can periodically evaluate the fire detection algorithms and their integration with the Rothermel Model. Each cycle provides an opportunity to enhance the model based on feedback and test outcomes.

Adaptability to New Developments: The iterative model is conducive to incorporating new datasets or adopting novel image processing and machine learning methodologies as they emerge, keeping your project up to date.

Step-by-Step Development: This approach allows you to concentrate on individual components of the project sequentially. For instance, you might initially focus on creating the fire detection algorithm, and thereafter work on integrating it with the Rothermel Model, refining each section progressively before advancing.

A diagram of a fire detection and processing

Description automatically generated

Figure 3. Detailed Processing Diagram of the Proposed Model

## 3.1 Data Acquisition, and Preprocessing

For my project I need to have a credible image dataset that I have extracted from Kaggle. I have used three separate image datasets and then combined them to one big dataset. My data collection and preprocessing process is given below.

A close-up of a sign

Description automatically generated

Figure 3. Detailed diagram for data preprocessing

Understanding the dataset

Reviewing and understanding any information or documentation attached with the dataset to deeply know the labels, source, content, and format of the respective data. Following that visualizing few samples images to get an idea of their quality. It is not necessary to have images of uniform sizes and same dimensions. Therefore, resizing is vital to ensure image consistency. And for that rescaling is required which can be done in Python.

Data Preprocessing

Data preprocessing is considered as one of the most vital initial stages of data mining because it converts rough data into a useable format. Data comes with many errors and inconsistencies; therefore, its preprocessing is crucial to make it clean and ready to use. A preprocessed data improves the accuracy, handles data imbalances, and transforms the data. For my project I have been using Python for data preprocessing.

*Loading Images*

Python provides a variety of libraries such as scikit-image and OpenCV, which are efficient in loading images from their respective paths.

*Image resizing*

For an image dataset, resizing is vital to get a consistent size for images. I am using Convolutional Neural Networks (CNN) which requires an input of a certain size.

*Data Augmentation*

Data augmentation is one of the efficient ways to increase image dataset diversity by augmenting the data in multiple angles, rotations, flipping, and zooming. This results in higher model accuracy and gives good results.

*Data labeling and encoding*

Data labeling and encoding is a crucial part of data preprocessing. My dataset is binary, so I have labelled using 0 and 1 where 0 is labelled for fire and 1 for no fire images respectively.

*Splitting data into testing and training*

To train a machine learning model efficiently it’s important to divide the data into training and testing. Training data is mostly 70-80% of the original data while the rest goes to the testing data.

### 3.1.1 Model selection and training

*Selecting a model*

Model selection is crucial for machine learning projects because they should give effective results based on their accuracy. Model selection is highly dependent on the type of dataset, for instance, some algorithms work well for image dataset while others don’t. I am dealing with images for my project, and it is binary classification, so I have chosen Convolution neural networks, Random Forest, and support vector machine.

*Model Training*

The dataset contains 70-80% training dataset which is used to train the model. However, the accuracy is dependent on the dataset quality therefore, rescaling and augmenting the data is vital for a higher accuracy. Below are the models I have trained for my project.

### 

### 

### 3.1.2 Standardization

Standardization is another feature scaling technique for data pre-processing. The purpose of standardization is to bring the dataset into a range that is close to the mean value of the data. The values of the feature will be changed so that, the mean became 0 and the standard deviation will be 1.[31]

For standardization, we can use the formula below,

(2)

Where X is the input value of a feature, – mean of the certain feature, and – standard deviation.

Eventually, two new data frames were created: normalized and standardized.

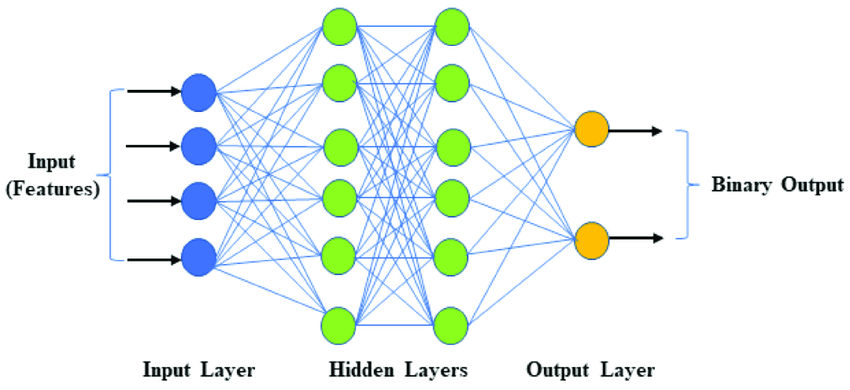
Last stage of data preprocessing module is data reduction. In this part, the feature selection has been done. After analyzing the correlation of the input variable to the target variable, some features have more impact on the output variables then others. Thus, the features with less impact were excluded from the dataset. With the dimension reduction only significant features were left, which is good for machine learning model.

## 3.2 Prediction

The last actual stage of the proposed model is prediction, in this stage we have used different machine learning algorithms for energy consumption prediction. Here, we will discuss some algorithms that maybe used for prediction.

### 3.2.1 Binary Convolution Neural Network

Binary Classification is one of the vital machine learning tasks which classifies data into two categories or classes. Binary classification using convolution neural networks makes the classification process easier and faster. The below diagrams depict how Binary CNN works.



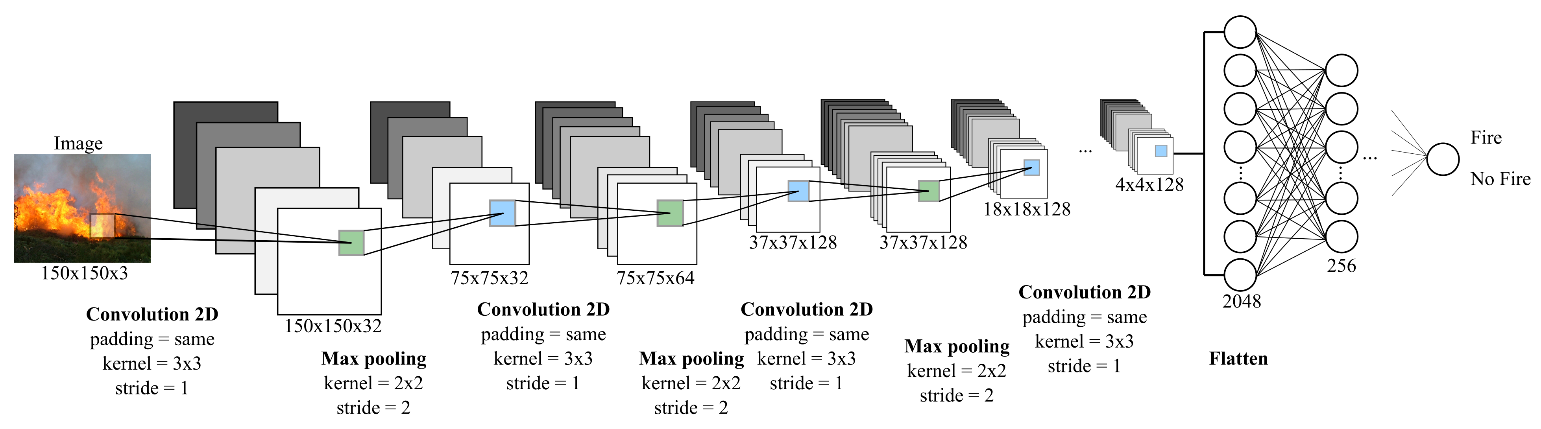


Figure 3. Schematic diagram for CNN

As showed in the Figure 3.3, in order to pass the data to hidden layer, the bias and weights should be summed.   
 (3)

– is an equation of Where *n* is number of inputs, *wi* denotes weight and b represents the bias term. In the hidden layer, activation functions like sigmoid, ReLU and tangent can be applied. These functions help to identify whether the input data is important or not for prediction [32].

Mathematically the sigmoid, ReLU, and tangent hyperbolic sigmoid action functions are represented in Equations (4-6) respectively.

(4)

() = max (0,) (5)

and for tangent activation function:   
 (6)

After applying activation functions, the algorithm passes all resulted values to output layer, where we get the result.

### 3.2.2 Random Forest Tree

Trees are not the ideal tool for predictive purpose because of their inaccuracy. Random Forest are also known as supervised tree-based machine learning algorithms, and they have their own merits as well. The figure below shows how a random forest model function.



Figure 3. Detailed diagram of Random forest tree

### 3.2.3 Support Vector Machine

Support Vector Machine (SVM) is a supervised machine learning algorithm which works efficiently for image classification. It is useful for both regression and classification purposes. The main objective of the SVM algorithm is to create a line or hyperplane that best segregates n-dimensional space, so in the future, we can easily put new data points in the right category. This best decision boundary is called a hyperplane.

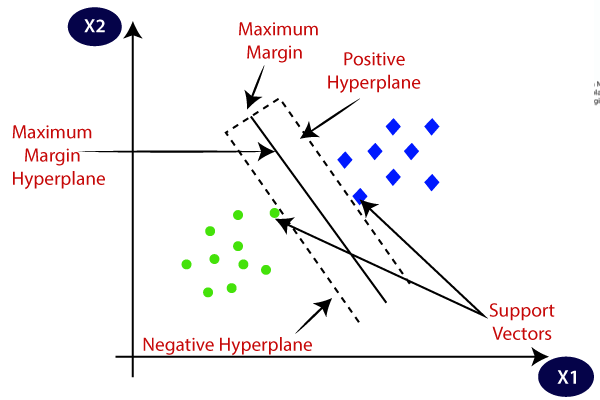


Figure 3. Schematic Diagram for SVM

### 3.2.4 ARIMA (MATHEMATICAL MODEL)

ARIMA, which is an efficient mathematical predictive model has been employed in R to make future forest fire analysis. Autoregressive Integrated Moving Average is a time series statistical analysis model that uses time series data, either to improve understanding of the set of data or predict future trends (Hayes, 2024).

A statistical model is autoregressive if it uses the past values in predicting the future ones. For example, an ARIMA model can predict stock prices based on a company's past performance or period-over-period foretelling of its earnings.

## 3.5 Application Module

Application module is followed by achieving excellent results of the trained models. This module serves the purpose of demonstrating the detection and prediction of forest fires. Streamlit written using python has been used to build the web-based application. The figure below shows how the application works. The user will be asked to create an account or log in with his existing credible credentials which will lead him to the dashboard. The dashboard comes with five tabs in the navigation bar where the home page will be displayed by default. The user can either upload or capture an image and then he can use the three models to detect the fire or no fire.

R shiny app has been integrated with the streamlit app which comes with four main features including finding fire spread rate using Rothermel model, creating a choropleth map, analyzing the Brazilian forest fires, and performing predictive analysis using ARIMA model.

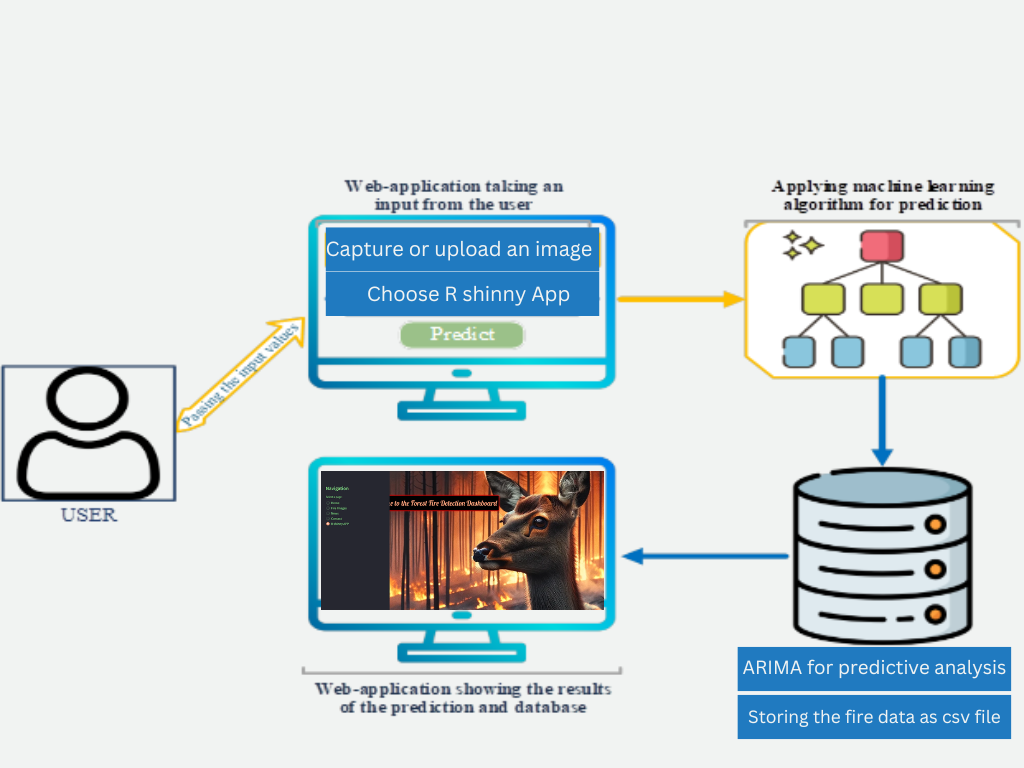


Figure 3. Illustration of web-application for prediction

# CHAPTER 4. IMPLEMENTATION, RESULTS, AND DISCUSSION

## 4.1 Implementation Setup

The implementation setup has been discussed briefly in this section. This project has been successfully carried out in 11th Gen Intel(R) Core (TM) i7-11800H @ 2.30GHz 2.30 GHz. The programing part includes Machine learning models written in Python programing language and mathematical models deployed in R programing language. The table below shows all the specifications.

Table 4. 1 Hardware and Software Specifications used in the proposed model implementations

|  |  |  |
| --- | --- | --- |
| **Type** | **Component** | **Details and Description** |
| Hardware | Computer type  Operating System  CPU  RAM | Katana GF66 11UD  Microsoft Windows 11 Pro  Intel(R) Core (TM) i7-11800H CPU @ 2.30GHz  16.0 GB DDR2 |
| Software | PyCharm  Jupiter Notebook  Matlab R2021b  R studio | Python 3.9.7  Python 3.9.7  MATLAB  R |
| Libraries | Tensorflow  Keras  Matplotlib  NumPy  Pandas  Sklearn | Used for building of neural networks  High-level powerful API that runs on top of Theano and Microsoft Cognitive Toolkit.  Used for interactive visualizations  Used for arrays and matrices for simplifying numerical operations  Used for data analysis  Used for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction. |
| Framework | Streamlit | Python Framework for building web-application |

## 4.1. Dataset

The project revolves around image classification, the images (fire and no fire) have been taken from Kaggle. For the fire analysis, a dataset of Brazilian forest fires has been used which is from Kaggle. The ARIMA model is used for predictive analysis or forest fire prediction where the data has been retrieved from NASA FIRMS using an API. The image dataset was augmented in python to make the dataset more credible and diverse.

## 4.2. Results

In the proposed project forest fire detection and prediction has been carried out using three different machine learning models and two mathematical models. The performance of models differs from each other and fluctuates according to the changing scenarios.

### 4.2.1. Random Forest Algorithm

The project is using an image dataset where the images have been divided into training, testing, and validation dataset. These three are further divided into fire and no fire images. So, we gave training images to train the random forest model and then tested it by testing images.

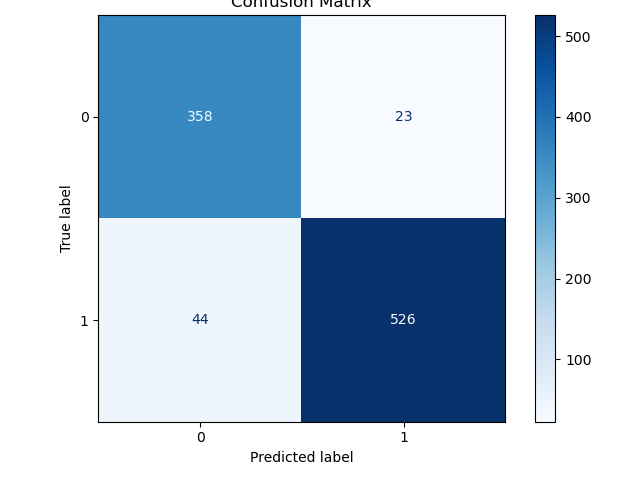


Figure 4. Fire and No Fire prediction confusion matrix using Random Forest.

The top left cell with 358 instances are true positive cases which are correctly classified as fire while 526 on the bottom right indicates the cases when the model correctly classified no fire images as no fire. The top right cell which has 23 shows false positives which mean that 23 cases were predicted incorrectly. The bottom left box with 44 instances shows false negatives which are the cases when fire images were predicted as no fire. Keeping these four factors into consideration accuracy, precision, Recall, and support has been calculated.

| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |  |
| --- | --- | --- | --- | --- | --- |
| 0 | 0.89 | 0.94 | 0.91 | 381 |  |
| 1 | 0.96 | 0.92 | 0.94 | 570 |  |

Figure 4. Performance evaluation of Random Forest.

### 4.2.2 Support Vector Machine

A screenshot of a graph

Description automatically generated

Figure 4. Confusion matrix for SVM.

The top left cell with 192 instances are true positive cases which are correctly classified as fire while 165 on the bottom right indicates the cases when the model correctly classified no fire images as no fire. The top right cell which has 0 shows false positives which mean that 0 cases were predicted incorrectly. The bottom left box with 25 instances shows false negatives which are the cases when fire images were predicted as no fire. Keeping these four factors into consideration accuracy, precision, Recall, and support has been calculated.

| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |  |
| --- | --- | --- | --- | --- | --- |
| 0 | 0.88 | 1.00 | 0.94 | 192 |  |
| 1 | 1.00 | 0.87 | 0.93 | 190 |  |

Figure 4. 4 Performance evaluation of SVM.

### 

### 4.2.3. Binary Convolution Neural Network

Binary convolution neural network works efficiently for image classification into two. This model is used when there are only two cases to predict. As the name suggests, convolution is the most vital part of the model which is used in the feature extraction.

Figure 4. 5 Confusion matrix for Binary CNN model

Figure 4. 6 Performance evaluation for Binary CNN

# CHAPTER 5. PERFORMANCE EVALUATION

In this section the performance evaluation of different algorithms that we have used in the proposed model has been carried out. For the performance evaluation we have used three basic metrices namely mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean square error (RMSE). The formulas for performance metrics are shown in

Equations (9-11):

(9)

(10)

and,

(11)

We have calculated these performances measures and listed in tables for each algorithm. We have also constructed graph of these performance metrices in order to better visualize the results.

Table 5. Performance evaluation of random forest algorithm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Statistical Meretrices** | **RMSE** |  | **MAE** | **MAPE** |
| Random Forest | 0.279 |  | 0.078 | 8.25% |
| Support Vector Machine | 69.49 |  | 48.46 | 22 |
| Binary Classification | 70.25 |  | 46.67 | 20 |

In this section we measured the performance of different machine learning algorithms by using different performance evaluators, such as root mean square error, mean absolute error, mean absolute percentage error. In this study we have considered simple and efficient machine learning algorithms, such as k-nearest neighbor algorithm, decision tree algorithm, artificial neural network algorithm, XGBoost algorithm, random forest, and cohesive model for energy consumption prediction. In most studies these algorithms have been applied either for short-term or long-term energy consumption prediction, but in this work, we have considered short-term, mid-term, and long-term and have applied the above-mentioned machine learning algorithms to predict energy consumption in building for these terms. We have not applied these algorithms directly on the data, first we preprocessed the data and removed missing values and made able for further processing, then on the preprocessed data the machine learning algorithm have been applied. We calculated the performance metrices of the results provided by the machine learning algorithms for short-term, mid-term and long-term energy consumption prediction. The results indicates that random forest performed well on the given data and the RMSE, MAE, and MAPE errors values are less than compared to others counterpart algorithms. Hence, the random forest algorithm is selected, and training model of this algorithms is then saved which is used in application layer. In the last module, which is application module, an interface is created, where a user can input specific parameters like time, day, month, and year, and can get the energy consumption perdition.

# CHAPTER 6. APPLICATION MODULE

The last module depicts how the project needs to be used by the users. The project has been deployed as a web application in the streamlit framework in Python and R Shinny using R. It consists of two parts such as machine learning part and mathematical part. The machine learning section consists of three trained models including SVM, RF, and Binary CNN to detect fire or no fire images either using uploaded image or captured image. On the other hand, the mathematical models include Rothermel and ARIMA model to calculate spread rate of forest fire and conduct predictive analysis respectively.

A screenshot of a login

Description automatically generated

Figure 6. Initial home page of the interface

The users have to create an account or login with the credible credentials to access the main dashboard, where the application sends all the data to the machine learning model. Out of all machine learning models tested in this project, Random Forest algorithm gave the best results of the prediction, however, other models have been added in the main dashboard as well. Next, the algorithms make a prediction based on the received values.

.



Figure 6. Dashboard page

After a successful login, the user will be able to access the main dashboard which is depicted above in the figure 6.2. The vertical navigation on the left hand side shows the main features of the web application. For the fire detection, the app gives two options either to upload an image or to capture an image. After successfully uploading or capturing the image the dashboard shows three buttons below the image labelled with the models. For instance, SVM, Random Forest, and Binary CNN. You can click on any model’s button and the result will be displayed on the screen. If it was a fire then an alert sound will be played as well.

A screenshot of a computer screen

Description automatically generated

Figure 6. Results display

The app has two parts one is the machine learning algorithms and another is comprised of mathematical models including rothermel for spread rate prediction and ARIMA model for doing predictive analysis. The navigation shows R shinny app feature where the mathematical models have been deployed in R programing language.

A screenshot of a computer

Description automatically generated

Figure 6. 4 R Shinny Dashboard

The R shinny app feature leads the user to the R shinny app whose interface looks like the above figure. The shinny app has five main features displayed vertically in the navigation bar. The features include fire spread rate, parameters, Choropleth map, forest fire analysis, and real time data. The spread rate is visualized as a 3D plot which changes with the changing parameters.

Diagram of a diagram of heat source

Description automatically generated

Figure 6. 7 Rothermel model equation

Wind and slop factor, moisture damping, and reaction intensity are referred to as heat source and they are vital parameters to determine the spread rate. While heat sink includes effective heating, heat of pre-ignition, and effective bulk density.

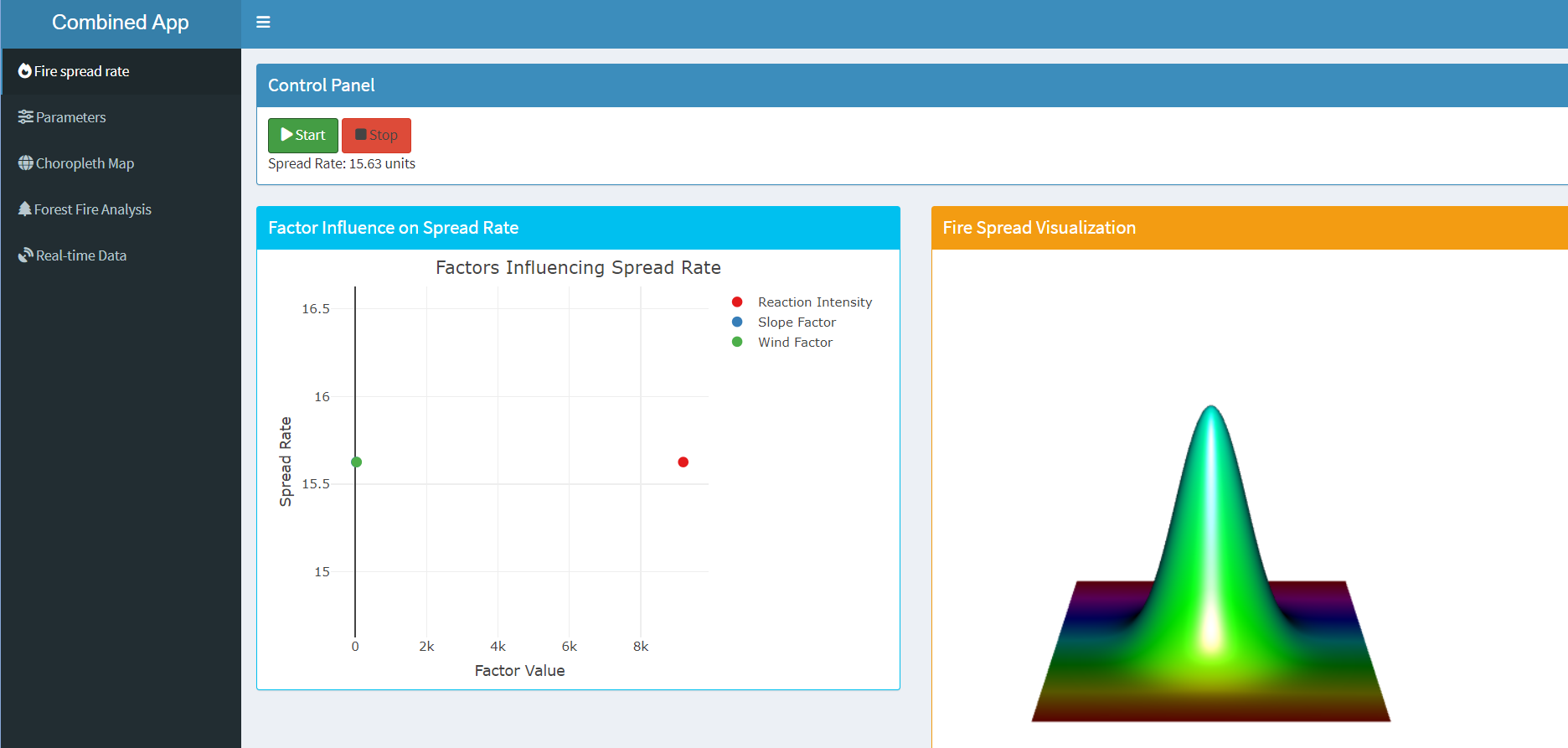


Figure 6. 8 Rothermel model visualization

The figure above shows the 3D plot visualization of rothermel model which changes with the parameters. Higher reaction intensity, wind factor, slope factor, and moisture damping results in a higher tail of the plot.

## 6.1 Similar applications comparison table

In an increasingly app-driven world, the marketplace teems with a multitude of applications, each vying for user attention and market dominance. Amidst this digital landscape, it becomes imperative to juxtapose similar apps, not only to discern their unique features and functionalities but also to understand how they meet different user needs or preferences. Whether it's for task management, fitness tracking, social networking, or productivity enhancement, a comparative analysis sheds light on the competitive edge of each app and highlights the innovations driving user engagement. As we delve into this comparison, we aim to unravel the nuances that set these applications apart, offering insights into their design, user experience, and overall performance. This exploration not only guides potential users in making informed decisions but also inspires developers to continuously enhance and differentiate their offerings in the crowded app ecosystem.

Table 6. 1 Similar app comparison table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Applications  Features | Firesmoke detection app | Firealarm inspection app | DRYAD | Fireangel connected app | Smoke detector inspection app | Nimbus | Your app | Total |
| Push notifications |  | Checkbox Checked outline | Checkbox Checked outline | Checkbox Checked outline | Checkbox Checked outline |  | Checkbox Checked outline | 5 |
| Detect both smoke and fire | Checkbox Checked outline |  | Checkbox Checked outline |  |  |  | Checkbox Checked outline | 3 |
| Using mathematical models |  |  |  |  |  |  | Checkbox Checked outline | 1 |
| send daily inspection reports |  | Checkbox Checked outline |  |  |  | Checkbox Checked outline | Checkbox Checked outline | 3 |
| Classifies the fire as severe, mild, and moderate |  |  |  |  |  |  | Checkbox Checked outline | 1 |

## 6.1.1 Links to applications

Fire smoke detection app (<https://store.azena.com/shop/p/A_00104000>)

Fire alarm inspection app (<https://axonator.com/micro-app-store/fire-alarm-inspection-app>) DRYAD (<https://www.dryad.net/>)

Fireangel connected app (<https://www.fireangel.co.uk/home/product/app/>)

Smoke detector inspection app (<https://www.fulcrumapp.com/apps/smoke-detector-inspection/>) Nimbus (<https://www.blazequel.com/videos/nimbus-fire-alarm-weekly-test-app/>)

# 6.2 Unique Features of my app

One of the features which makes my app unique is using mathematical models (Rothermel model) to predict the forest fires using real time data from the NASA FIRMS. The ARIMA (Autoregressive Integrated Moving Average) prediction model has been used to predict forest fires using the respective data. The Rothermel model finds the spread rate of fire using the given data and produces a choropleth map for it. Mathematical models are highly efficient in prediction related projects therefore, I chose them.

# 6.3 Technical specifications of the project

**Note**: Write a clear technical specification. Specify all functional and non-functional requirements of your project. For all requirements provide a description and indicate acceptance criteria.

Table 6.2 Functional requirements

|  |  |
| --- | --- |
| **Functional**  **requirements** | **Functional Requirement** |
| **FR 1** | Preprocessing the images and their acquisition |
| **FR 2** | Algorithm for fire detection |
| **FR 3** | Real-time monitoring of the respective area |
| **FR 4** | Notifications and alerts |

Table 6.3 Non-functional requirements

## 

|  |  |
| --- | --- |
| **Non-Functional requirement** | **Functional Requirement** |
| **NFR 1** | Preprocessing the images and their acquisition |
| **NFR 2** | Algorithm for fire detection |
| **NFR 3** | Real-time monitoring of the respective area |
| **NFR 4** | Notifications and alerts |

# 

Here is the detailed budget description that I need for the completion of my project.

Table 6.4 Project Budget Estimation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Category* | *Description* | *Quantity* | *Unit price* | *Total cost* | *% Of Budget* |
| Deployment | 1TB external hard disk | 10 months | $ 100 | $ 100 | 33.3% |
| Plagiarism report | Two times document check payment with plagiarism checker[[1]](#footnote-1) | 2 times | $ 35 | $ 70 | 23% |
| equipment | 16GB RAM | 1 | $90 | $90 | 30% |
| Online course(streamlit) | Making the UI of the app using streamlit | 1 | $20 | $20 | 6.66% |
| Unexpected expenses | Unforeseen expenses | 1 | $20 | $20 | 6.66% |
| Total Budget Cost Estimate = $300 | | | | |  |

# 

# 

# 6.4 Project plan and schedule

**Note**: Provide the project schedule using the Gannt Chart and comment where applicable. Identify milestones. A milestone is a concrete event that one can use to demonstrate progress. Milestones should be clear, concrete, demonstrable achievements (“SMART”).

Table 6.5 Milestones of my project

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S/N | WEEK | To be done | MILESTONE | OUTCOMES |
| 1 | 2 Weeks | 22/10/2023 | Project initiation and planning | Clear objectives and scope of the project. |
| 2 | 4-6 Weeks | 30/11/2023 | Dataset acquisition and preparation | Collect project related dataset such as drone or satellite images. |
| 3 | 2 Weeks | 14/12/2023 | Cleaning, labelling, and annotating the data | Will get a clear picture of the dataset, because the outliers will be removed. |
| 4 | 2 Weeks | 2/01/2024 | Model selection and development | Selecting the relevant and appropriate machine learning model. |
| 5 | 4 Weeks | 2/02/2024 | Model training | The model will be trained based on the final dataset. |
| 6 | 4 Weeks | 3/03/2024 | Testing and validating the model | Analyze the performance of the model using various datasets. |
| 7 | 2 Weeks | 28/03/2024 | Deploying the model | Deploying the model at UCA |
| 8 | 3 Weeks | 20/04/2024 | Documentation | Write a report based on the whole project. |
| 9 | 1 Weeks | 30/04/2024 | Revision | Revising the whole project and the documentation. |

**6.4.1 Work Breakdown of project**

A diagram of a project planning

Description automatically generated

Figure 6.1 Work breakdown

## 6.4.2 Gannt Chart

A calendar with colorful labels

Description automatically generated with medium confidence

Figure 6.2 Ghantt Chart of the project

# 

# 

# 

# 

# 

# 

# 

# 

# 

# CHAPTER 7. RISK MANAGEMENT AND PLAN

**Note**: Identify your risks and solutions to them. The quality of the plan will impact your midterm exam grade.

Table 7.1 Technical Risks Assessment Table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Risks | | Likelihood | Consequence | Overall Risk | Risk Level |
| (1-3) | (1-5) | (1-15) |
| 1 | unbalance dataset | 1 | 4 | 10 | Major |
| 2 | Model Sturdiness | 2 | 4 | 11 | Major |
| 3 | Model Fairness | 3 | 5 | 13 | Severe |
| 4 | Equipment malfunctioning | 1 | 1 | 1 | insignificant |

* 1. **Four main risks**

## 7.1.1 The quality and quantity of data (Balance Data)

*Description*

Training data is one of the most vital aspects of a machine learning trained model. The dataset should be unbiased and sufficient to have accurate predictions and results. Having a balanced dataset is equally important to avoid false results. For instance, having more fire pictures than non-fire pictures lead to inaccurate results.

*Solution*

Will get the dataset from a reliable and legitimate source. The images present in the dataset should be of good quality and updated. Fire images, non-fire images, and smoke images should be of equal number.

## 7.1.2 Model Sturdiness and adaptability.

*Description*

Over fitness occurs when a model works perfectly for a certain data set and doesn’t perform well on a new data set. Overfitted models are indeed not efficient for fire detecting models.

*Solution*

To avoid overfitted model, I will frequently retrain the model on fresh dataset.

## 7.1.3 Model Fairness and Biasness

*Description*

Models can never be 100% accurate because they are trained using certain data. The data might not be well balance and updated which makes the model biased.

*Solution*

To have an unbiased model, I will take updated and balanced dataset of images. I will acquire data from multiple sources and then compile them to make my final dataset.

## 7.1.4 Equipment Malfunctioning

*Description*

The equipment that I am using, such as my laptop may malfunction due to various reasons such as some software or hardware issues.

*Solution*

There are multiple PCs in the CS lab and library of UCA that I can use to work.

## CHAPTER 8. DATABASE DIAGRAMS

Database diagram is a visual representation of the database of structure of the application. It provides the blueprint for the design for the understanding of the database and is a very important for representation of any project.

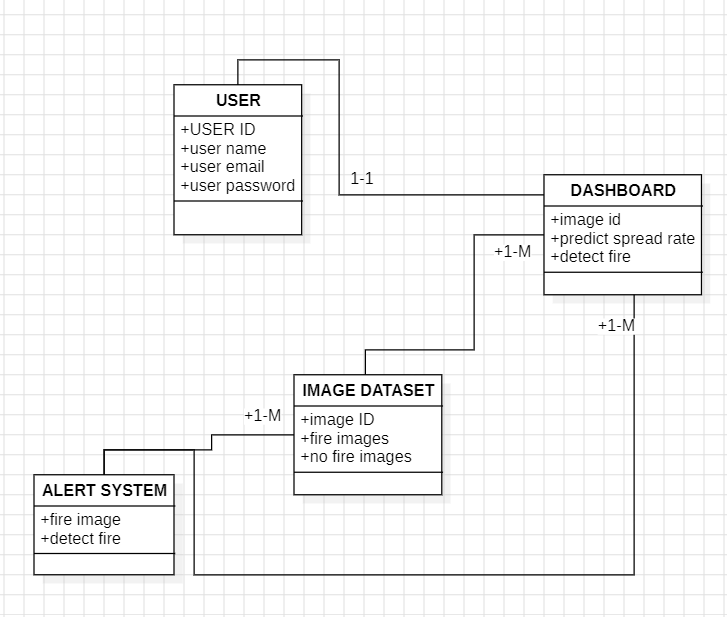


Figure 8.1

A diagram of a data flow

Description automatically generated

Figure 8.2

A diagram of a software company

Description automatically generated

Figure 8.3

**Conclusion**

# References

A, M. (2018, April 27). Wildland fires detection and segmentation using deep learning. *Journal of applied remote sensing, 10649*, 12-26. doi:https://doi.org/10.1117/12.2304936

Abdusalomov et al. (2023). An Improved Forest Fire Detection Method Based on the Detectron2 Model and a Deep Learning Approach. *Sensors, 23*(3), 20-45. doi:https://doi.org/10.3390/s23031512

Abdusalomov, A. B. (2023). An Improved Forest Fire Detection Method Based on the Detectron2 Model and a Deep Learning Approach. *Sensors, 23*(3). doi:https://doi.org/10.3390/s23031512

Alkhatib, R. (2023, July 13). A Brief Review of Machine Learning Algorithms in Forest Fires Science. *Appl. Sci*. doi:https://doi.org/10.3390/app13148275

Benzekri, W. (2020). Early Forest Fire Detection System using Wireless. *International Journal of Advanced Computer Science and Applications, 11*(5), 1-8. Retrieved from https://pdfs.semanticscholar.org/fca3/1181e24a3148661e20d4e20d81ac9b06a426.pdf

*Cambridge reference*. (n.d.). Retrieved from https://dictionary.cambridge.org/: https://dictionary.cambridge.org/dictionary/english/forest-fire

Dampage, U. (2022). Forest fire detection system using wireless sensor networks and machine learning. *Scientific Reports*. doi:https://doi.org/10.1038/s41598-021-03882-9

Elshewey, A. M. (2020). Forest Fires Detection Using Machine. *Journal of Xi'an University of Architecture & Technology, 8*(9), 1-8. Retrieved from https://www.researchgate.net/profile/Amira-Elsonbaty-2/publication/344462171\_Forest\_Fires\_Detection\_Using\_Machine\_Learning\_Techniques/links/5f78c3f6299bf1b53e0c1c1e/Forest-Fires-Detection-Using-Machine-Learning-Techniques.pdf

Fan, R. (2021). Lightweight Forest Fire Detection Based on Deep Learning. *IEEE*, 1-6. doi:10.1109/MLSP52302.2021.9596409

firelab. (2023, March 3). *The Rothermel Fire Spread Model: A 50-year milestone in fire research.* Rocky Mountain Research Station. firelab. Retrieved from https://www.firelab.org/news/rothermel-fire-spread-model-50-year-milestone-fire-research

Hayes, A. (2024, February 23). Autoregressive Integrated Moving Average (ARIMA) Prediction Model. *Investopedia*, 1-5. Retrieved from https://www.investopedia.com/terms/a/autoregressive-integrated-moving-average-arima.asp#:~:text=An%20autoregressive%20integrated%20moving%20average%2C%20or%20ARIMA%2C%20is%20a%20statistical,values%20based%20on%20past%20values.

Joosse, T. (2020, December 8). Human-sparked wildfires are more destructive than those caused by nature. *Science*, 1-9. doi:10.1126/science.abg0529

Kanga, Y. (2022, November 16). A deep learning model using geostationary satellite data for forest fire detection with reduced detection latency. *GIScience & Remote Sensing, 59*(1), 2019-2035 . doi:https://doi.org/10.1080/15481603.2022.2143872

Khan, A. (2022, April 28). DeepFire: A Novel Dataset and Deep Transfer Learning Benchmark for Forest Fire Detection. *Hindawi*. doi:h10.1155/2022/5358359

Kinaneva, D. (2019, May 20). Early Forest Fire Detection Using Drones and Artificial Intelligence. *MIPRO*, 1060-1065. doi:https://doi.org/10.23919/MIPRO.2019.8756696

KUKUK, S. B. (2021, July). Comprehensive Analysis of Forest Fire Detection using Deep Learning Models and Conventional Machine Learning Algorithms. *International Journal of computational and experimental science and engineering, 7*(2), 84-94. doi:https://doi.org/10.22399/ijcesen.950045

MacCarthy et al. (2023, August 29). *The Latest Data Confirms: Forest Fires Are Getting Worse*. Retrieved from world resource institute: https://www.wri.org/insights/global-trends-forest-fires

Peng, Y. (2019, December). Real-time forest smoke detection using hand-designed features and deep learning. *Computers and Electronics in Agriculture, 167*. doi:https://doi.org/10.1016/j.compag.2019.105029

Pragati. (2020). FOREST FIRE DETECTION USING MACHINE LEARNING. *INTERNATIONAL JOURNAL OF ADVANCE SCIENTIFIC RESEARCH, 4*(12), 1-7.

priya, R. S. (2019). Deep Learning Based Forest Fire Classification and Detection in Satellite Images. 61-65. doi:10.1109/ICoAC48765.2019.246817

Qingjie, Z. (2016, January ). Deep Convolutional Neural Networks for Forest Fire Detection. *Springer nature*. doi:https://doi.org/10.2991/ifmeita-16.2016.105

Rajan et al. (2022, Januray 12). FOREST FIRE DETECTION USING MACHINE LEARNING. 1-10. Retrieved from https://www.researchgate.net/publication/357811123\_FOREST\_FIRE\_DETECTION\_USING\_MACHINE\_LEARNING

Rajan, G. V. (2022, January ). FOREST FIRE DETECTION USING MACHINE LEARNING. Retrieved from https://www.researchgate.net/publication/357811123\_FOREST\_FIRE\_DETECTION\_USING\_MACHINE\_LEARNING

Sathishkumar et al. (2023, February 17). Forest fire and smoke detection using deep learning-based learning without forgetting. *Fire Ecology, 19*(9), 12-27. doi:https://doi.org/10.1186/s42408-022-00165-0

Sathishkumar, V. E. (2023). Forest fire and smoke detection using deep learning-based learning without forgetting. *Fire Ecology, 9*, 12-78. doi:https://doi.org/10.1186/s42408-022-00165-0

Seydi et al. (2022, February 21). Fire-Net: A Deep Learning Framework for Active Forest Fire Detection. *Journal of sensors, 2022*, 24-70. doi:https://doi.org/10.1155/2022/8044390

Seydi, S. T. (2022). Fire-Net: A Deep Learning Framework for Active Forest Fire Detection. *Hindawi*. doi:https://doi.org/10.1155/2022/8044390

Sobha, P. (2023, July). A Survey of the Machine Learning Models for Forest Fire Prediction and Detection. *International Journal of Communications, Network and System Sciences, 16*(7). doi:https://doi.org/10.4236/ijcns.2023.167010

*WHO*. (n.d.). Retrieved from World Health organization: https://www.who.int/health-topics/wildfires#tab=tab\_1

Xu, R. (2021). A Forest Fire Detection System Based on Ensemble Learning. *Forests, 12*(2). doi:https://doi.org/10.3390/f12020217

1. [↑](#footnote-ref-1)