Machine Learning based Wildfire Prediction & Analysis

Submitted in partial fulfillment of the requirements of the degree

Bachelor Of Engineering in **Computer Engineering**

By

Abhishek Goud (2203050)

Akshay Kadam (2203068)

Ayush Kamath (2203071)

Saurav Jagmalani (2104063)

Supervisor
Dr. Ujwala Bharambe

Computer Engineering Department

Thadomal Shahani Engineering College

Bandra(W), Mumbai - 400 050

University of Mumbai

(AY 2024-25)

CERTIFICATE

This is to certify that the Mini Project entitled "**Machine Learning based Wildfire Prediction & Analysis**" is a bonafide work of **Abhishek Goud (2203050**), **Akshay Kadam (2203068)**, **Ayush Kamath (2203071)**, **Saurav Jagmalani (2104063)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of "Bachelor of Engineering" in "Computer Engineering".

Dr. Ujwala Bharambe
Supervisor

Dr. Tanuja Sarode
(Head of Department)

Dr. G. T. Thampi
(Principal)

Mini Project Approval

This Mini Project entitled **“Machine Learning based Wildfire Prediction & Analysis”** By **Abhishek Goud(2203050)**, **Akshay Kadam (2203068), Ayush Kamath (2203071), Saurav Jagmalani (2104063)** is approved for the degree of **Bachelor of Engineering** in **Computer Engineering.**

Examiners

1…………………………………………
(Internal Examiner Name & Sign)

2…………………………………………
(External Examiner name & Sign)

Date:

Place:

Contents

Acknowledgement

We would like to express our gratitude and thanks to **Dr. Ujwala Bharambe** for her valuable guidance and help. We are indebted for her guidance and constant supervision as well as for providing necessary information regarding the project. We would like to express our greatest appreciation to our principal **Dr. G.T. Thampi** and head of the department **Dr. Tanuja Sarode** for their encouragement and tremendous support. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of the project.

Abhishek Goud (2203050)

Akshay Kadam (2203068)

Ayush Kamath (2203071)

Saurav Jagmalani (2104063)

1. Introduction

With wildfires becoming increasingly frequent and destructive, early detection and timely response are more critical than ever. This project, "Machine Learning-Based Wildfire Prediction and Analysis" presents a real-time monitoring system powered by machine learning. Using satellite imagery and weather data, the platform predicts and tracks wildfire activity. Built with React and Flask, it offers an accessible, data-driven solution to help both authorities and the public respond quickly and effectively.

1.1 Introduction

The wildfire monitoring website is designed to provide real-time detection and tracking of wildfires using advanced machine learning (ML) techniques. With the increasing frequency and severity of wildfires worldwide, early detection has become crucial in mitigating the devastating impact on both natural ecosystems and human settlements. This project aims to leverage modern technologies such as React for the frontend and Flask for the backend, combined with a machine learning model capable of analyzing satellite images and weather data to predict and detect wildfire occurrences.

The platform will allow users to access real-time information about ongoing wildfires, track the progression, and receive updates on potential risks. By integrating a user-friendly interface with powerful backend processing, the system will streamline the analysis and presentation of large datasets, making it accessible for both government agencies and the general public. Through this solution, we hope to contribute to better disaster preparedness and quicker response times, ultimately helping to minimize the damage caused by wildfires.

1.2 Motivation

The motivation behind this project stems from the increasing global threat posed by wildfires. Over the past few decades, the frequency, intensity, and scale of wildfires have surged due to climate change, deforestation, and human activity. These fires not only destroy vast areas of forests, wildlife habitats, and biodiversity but also put human lives and properties at great risk. Traditional methods of wildfire detection often rely on manual reporting or satellite systems that lack real-time analysis, leading to delays in response and containment efforts.

By developing a wildfire monitoring system that leverages machine learning and modern web technologies, this project aims to improve early detection, providing real-time insights to authorities and the public. The integration of data-driven models with a user-friendly interface empowers communities and decision-makers to act quickly, reducing the environmental, economic, and human toll of wildfires. This project represents an effort to harness the power of technology for a critical global issue, offering a solution that could enhance wildfire management and disaster preparedness.

1.3 Problem Statement & Objectives

This project focuses on developing a real-time wildfire monitoring website that uses machine learning for early detection and tracking of wildfires, aiming to minimize the impact on natural resources and human lives.

Objectives:

- Implement a machine learning model to detect and predict wildfires using satellite images and weather data.
- Develop a user-friendly web interface with React for real-time wildfire tracking and information dissemination.
- Integrate Flask as the backend for data processing and API management, ensuring seamless communication between the frontend and the Machine Learning model.
- Deploy the system on cloud platforms to enable real-time access and scalability for users.

1.4 Organization of the Report

- Introduction:

The first chapter introduces the topic of wildfire monitoring, outlining the problem statement, the motivation behind developing the system, and the objectives of the project. It emphasizes the importance of real-time detection and monitoring of wildfires to mitigate their destructive impact on the environment and human life.

- Literature Survey:

The second chapter covers the literature survey, which includes all the research work conducted in relation to wildfire detection and monitoring. It discusses existing systems, technologies, and models used in similar projects. This chapter also includes information about the learning of new tools and techniques, such as machine learning algorithms for wildfire prediction, data processing tools, and web development frameworks (React, Flask) that are relevant to the project.

- Proposed System:

The third chapter details the proposed wildfire monitoring system developed in this project. It includes the block diagram, the techniques employed for detection, and a description of the hardware and software components used. Screenshots of the system interface and the working model are presented to demonstrate the system's functionality.

2. Literature Survey

2.1 Review of Existing Systems

A thorough review of existing wildfire monitoring systems is essential to identify current capabilities, limitations, and opportunities for improvement. Several systems and applications have been developed to detect, monitor, and report wildfire activity using satellite data, ground-based sensors, and integrated alert mechanisms. These platforms vary in terms of geographic coverage, data sources, user interface, and functional features. This section presents an overview of some widely used wildfire monitoring systems and highlights their key characteristics, serving as a foundation for the development of an improved, predictive solution.

System	Description	Key Features
NASA's FIRMS	Provides real-time active fire data and hotspots using MODIS satellites.	Real-time fire detection, global maps, historical fire trends, data visualization tools.
Wildfire Tracker App	Offers real-time wildfire updates including location, size, and containment status from government sources.	Interactive maps, push notifications, safety tips, localized alerts.
Global Forest Watch Fires	Combines satellite imagery and ground reports for real-time global wildfire monitoring.	Fire hotspot maps, risk assessments, ecosystem impact analysis, historical data access.
FireMapper	A web-based tool for visualizing and monitoring wildfires using satellite data and fire behavior models.	Real-time updates, fire spread modeling, interactive maps, historical records.
California Wildfire App	Focused on California, this app provides real-time alerts, safety resources, and fire-related data from CAL FIRE and local agencies.	Evacuation routes, air quality updates, safety alerts, community resources.

Table 2.1.1

2.2 Research Gaps in Current ML-Based Approaches

While several wildfire monitoring and alert systems are currently in use, they exhibit a range of limitations that reduce their overall effectiveness, especially in critical situations. These limitations affect their scalability, responsiveness, user accessibility, and predictive accuracy. A detailed analysis of these shortcomings helps identify gaps in the existing approaches and highlights the need for more intelligent, inclusive, and proactive wildfire management solutions. The table below summarizes the key limitations observed in current wildfire monitoring systems.

S. No.	Limitation	Description
1	Limited Geographic Coverage	Many systems are region-specific (e.g., California Wildfire App), making them less useful for users in other states or countries.
2	Data Latency	Systems like NASA's FIRMS and Global Forest Watch Fires depend on satellite data, which may have delays between fire occurrence and detection.
3	User Interface and Experience	Some applications have complex or unintuitive user interfaces, making it difficult for users to quickly access critical information during emergencies.
4	Dependence on External Data Sources	Relying on third-party agencies for data can lead to inconsistencies and delays in updates if the data providers do not report in real time.
5	Limited Predictive Capabilities	Most existing systems focus on real-time monitoring and lack machine learning-based prediction models for forecasting potential wildfire outbreaks.
6	Insufficient Community Engagement	Current systems often do not include features for community reporting or participation, missing valuable local insights and real-time user feedback.
7	Environmental Factors Ignored	Many platforms overlook associated environmental impacts such as air quality, smoke dispersion, and ecological consequences.
8	Technical Limitations	Systems often require constant internet access, which can be a challenge in remote or wildfire-affected areas with poor connectivity.

Table 2.2.1: Limitations of Existing Wildfire Monitoring Systems

2.3 Contribution of the ML-Based Mini Project

In developing our wildfire monitoring system, we aimed to address several limitations identified in existing solutions while introducing unique features that enhance its effectiveness and user experience. Here's how our project differs and improves upon current systems:

S. No.	Contribution	Description
1	Real-Time Data Processing	Utilizes a machine learning model to process data in real-time, enabling faster detection and immediate alerts compared to systems with satellite data latency.
2	Comprehensive Geographic Coverage	Designed for global use, incorporating diverse public datasets to ensure relevance across different regions, unlike region-specific existing systems.
3	User-Centric Interface	Built with React, the interface emphasizes simplicity and ease of use, allowing quick access to vital information, especially during emergencies.
4	Predictive Analytics Integration	Incorporates ML algorithms for forecasting wildfires based on weather, historical patterns, and environmental data, enabling proactive response and planning.
5	Community Engagement Features	Allows users to report local fire incidents and observations, promoting crowdsourced data collection and enhancing local awareness and participation.
6	Environmental Impact Monitoring	Includes features for monitoring air quality and smoke dispersion, offering users a more complete picture of wildfire-related environmental effects.
7	Offline Access and Notifications	Supports offline functionality and push notifications to ensure critical updates reach users in areas with limited or no internet connectivity.
8	Scalable Architecture	Developed using Flask and React, the system architecture is scalable to accommodate more users and data without compromising performance.

Table 2.3.1

3. Proposed System

This chapter consists of detailed description about the methodology used, the hardware and software components, the tools used and also the screenshots of the project.

3.1 Overview of the Proposed ML System

This project leverages a comprehensive stack of geospatial tools and machine learning frameworks to detect and assess wildfires using satellite imagery. At its core, the system is built on Google Earth Engine (GEE), which serves as the primary platform for sourcing and processing satellite data, including Sentinel-2, FireCCI, and ERA5 collections. Python APIs are used to extract time-series features such as NDVI, land surface temperature, wind vectors, and burn indicators at multiple geographic points across California.

To ensure secure and efficient interaction with GEE, the system authenticates using service account credentials, allowing programmatic access to Earth Engine's extensive datasets. The collected data is preprocessed and exported as CSV files to Google Drive, preserving the spatial and temporal granularity necessary for training robust detection models.

Further, machine learning which is implemented separately, consumes this data for training wildfire detection and confidence regression models. Key features such as NDVI (derived from Sentinel-2 bands B8 and B4), temperature, wind speed/direction (from ERA5), and fire-specific attributes like burn date and confidence (from FireCCI) are extracted without aggregation to retain the native image-level variation.

The project emphasizes both scale and precision, targeting the generation of a large, diverse dataset (10,000+ samples) that reflects California's wildfire patterns from 2016 to 2024. This data-driven approach lays the foundation for building interpretable and generalizable models capable of early wildfire detection and risk quantification.

3.2 System Architecture for Wildfire Prediction

Regression predictor architecture:

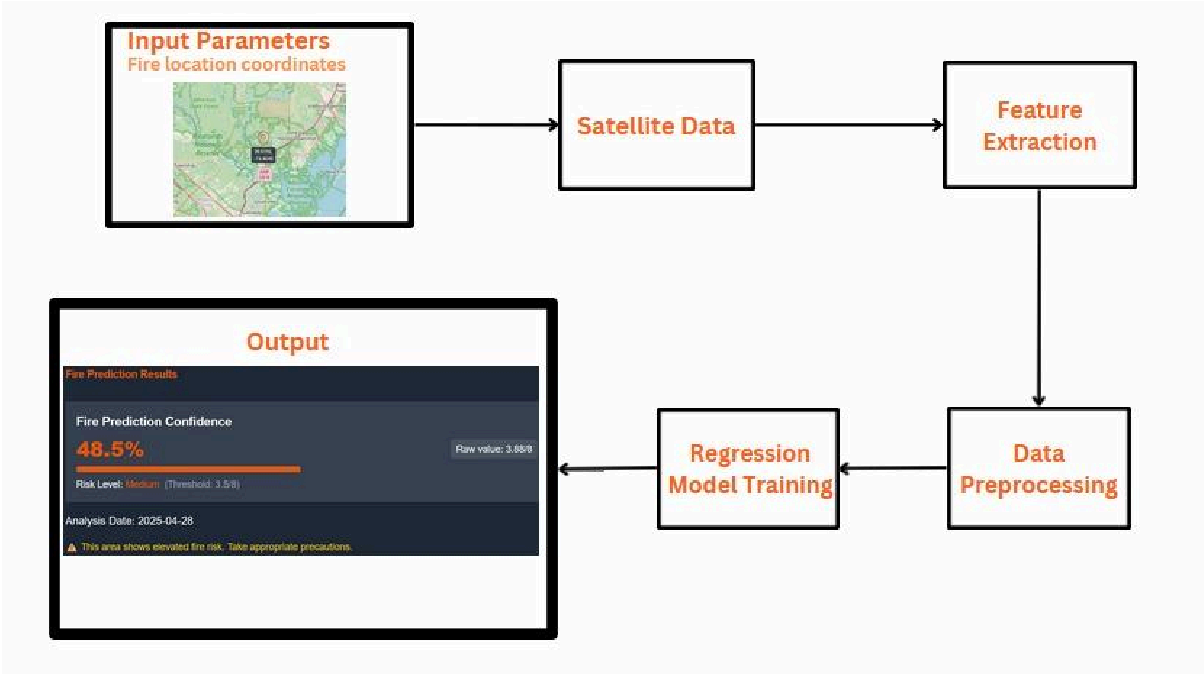

Image 3.2.1

The regression predictor architecture is developed to estimate wildfire risk based on location-specific environmental features derived from satellite data. Initially, fire location coordinates are ingested as input parameters, which trigger the acquisition of corresponding satellite-derived datasets.

Following data acquisition, a feature extraction module isolates key predictive attributes, including vegetation indices, land cover classifications, surface temperature, and atmospheric wind vectors. Extracted features undergo a comprehensive data preprocessing phase involving normalization, encoding, and outlier treatment to ensure model readiness.

Preprocessed data is subsequently fed into a regression model training pipeline, employing algorithms such as Elastic Net, Random Forest, and XGBoost to learn complex mappings between input features and fire risk confidence levels. The trained model produces a quantitative fire prediction confidence score, dynamically mapped to categorized risk levels (e.g., low, medium, high) based on calibrated thresholds.

The overall architecture emphasizes modularity, scalability, and adaptability to varying spatial and temporal resolutions of wildfire-related predictors, ensuring flexibility in operational deployment.

Fire spread flowchart:

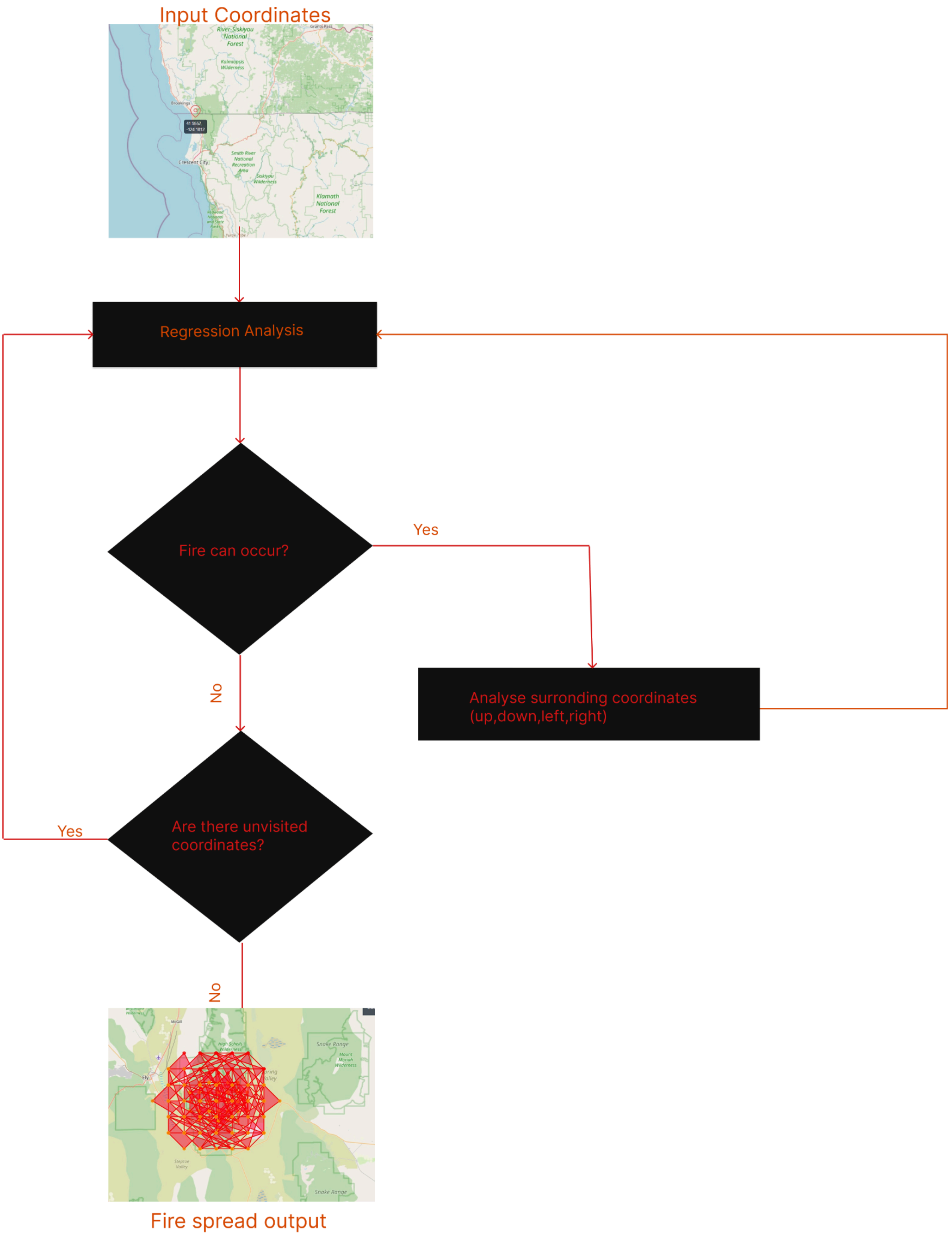

Image 3.2.2

The proposed system architecture for wildfire prediction and spread simulation is designed as a modular, data-driven pipeline that integrates satellite observations, environmental feature analysis, and machine learning models to forecast potential fire-prone zones with high accuracy. The system's design emphasizes scalability, modularity, and real-time responsiveness, which are critical for operational deployment in wildfire management systems.

The process initiates with the ingestion of essential input parameters provided by the user, specifically the geographic coordinates of the initial fire location and the corresponding time of observation. These parameters serve as the primary anchors for the retrieval and contextualization of environmental data.

Upon receiving the input, the system queries trusted satellite data repositories such as FireCCI, ERA5, and Sentinel-2. These sources provide comprehensive geospatial datasets encompassing critical variables like land surface temperature, vegetation health indicators (e.g., NDVI—Normalized Difference Vegetation Index), surface wind vectors, and other climatic factors influential to wildfire behavior. The raw satellite data is subjected to a dedicated feature extraction module, where relevant environmental attributes are derived to construct the predictive feature set.

Following feature extraction, a robust data preprocessing phase is employed to enhance data quality and model compatibility. This phase involves addressing missing data, performing noise reduction, and normalizing feature scales to ensure homogeneity across input variables. Such preprocessing steps are essential to mitigate bias and variance issues during model training and inference.

At the core of the system lies a regression-based machine learning model, trained on extensive historical fire incident datasets. This model receives the preprocessed feature vector and outputs a probability score reflecting the likelihood of fire occurrence at the specified location. If the predicted probability surpasses a calibrated threshold, the system recognizes a potential fire event and subsequently initiates a recursive spatial analysis of neighboring coordinates (north, south, east, and west).

The fire spread simulation is structured as an iterative procedure, where each newly identified coordinate undergoes the same sequence of feature extraction, preprocessing, and fire risk prediction. This recursive process continues until no further unvisited neighboring coordinates meet the fire occurrence criteria. The design effectively mimics the natural spread of wildfires by accounting for localized environmental conditions influencing propagation dynamics.

The final output consists of an aggregated set of geographic coordinates predicted to be at risk. These results are visualized as a spatial fire spread map, offering critical insights for early warning systems, emergency response planning, and resource allocation strategies. Overall, the architecture integrates predictive analytics with dynamic simulation, making it a powerful tool for wildfire risk assessment and spread forecasting.

3.3 Algorithm and Process Design

Formulating the Problem Statement:

The foundational objective of this project is to formulate and solve a regression problem aimed at predicting wildfire detection confidence levels based on a combination of satellite-derived and meteorological variables. Instead of framing the task as a binary or categorical classification problem, we explicitly model it as a supervised regression task where the goal is to predict a continuous target variable-Confidence level-associated with the likelihood of a fire being correctly detected at a given geospatial location and time.
This formulation allows the model to provide nuanced, real-valued confidence predictions, which can be more informative than binary outputs in applications such as early wildfire warning systems, dynamic resource allocation, and fire suppression planning.
The dataset is structured as follows:

Feature Name	**Description**	**Variable Notation**
Burn Date	Date when the fire was detected by FireCCI	X1
NDVI	Normalized Difference Vegetation Index from Sentinel-2	X2
Observed Fire Flag	Binary flag indicating whether a fire was observed	X3
Mean 2m Air Temperature	Average near-surface temperature from ERA5	X4
U-component of Wind	Zonal wind velocity at 2 meters (west-east)	X5
V-component of Wind	Meridional wind velocity at 2 meters (south-north)	X6
	Target: Confidence Level (FireCCI)	y

$$Y = b1*X1 + b2*X2 + b3*X3 + b4*X4 + b5*X5 + b6*X6 + b7*X7 + a$$

Each instance in the dataset is a timestamped geospatial observation, represented as a tuple (X1, X2,…,X6;y), where the features capture both environmental and fire-specific context. These tuples are extracted across multiple dates and locations within California from 2016 to 2024, ensuring both temporal and spatial diversity in the training data.
By explicitly framing the problem in this way, we enable the development of regression models capable of learning complex, non-linear relationships between environmental conditions and fire detection certainty. This not only enhances prediction accuracy but also improves interpretability and practical deployment in real-time systems.

Understanding the Framework and Requirements:

With the problem clearly defined, the next step involved outlining the technical architecture and functional requirements of the system. Given the temporal and spatial dimensions of the input data, the framework needed to support scalable extraction, transformation, and storage of image-level features across multiple time steps and locations. Google Earth Engine (GEE) was selected as the backbone for geospatial data retrieval and preprocessing. The system also required integration with Google Drive for persistent storage of large CSV exports. Functionally, the pipeline needed to aggregate values monthly in order to work with the limited hardware without compromising integrity of per-image observations, in order to retain high-resolution temporal variability that would inform the regression model. Emphasis was placed on flexibility, allowing new features or temporal windows to be incorporated without significant architectural modifications.

Identifying Tools and Technologies to be Used:

To meet the outlined requirements, the project incorporated a modular set of technologies. GEE's JavaScript and Python APIs were used to programmatically access and extract relevant satellite data from the FireCCI51, Sentinel-2, and ERA5-Land datasets. Google Drive was used as a temporary data sink to facilitate structured CSV exports. For environmental variables, the Sentinel-2 imagery provided spectral information for calculating NDVI, while the ERA5-Land reanalysis datasets contributed meteorological parameters such as temperature and wind vectors. FireCCI51 was used to extract wildfire-specific indicators, including burn date, observed fire flag, and confidence level. The collected features were aligned temporally and spatially using Python-based geospatial libraries and subsequently stored in structured formats compatible with downstream ML workflows. The final dataset was designed to accommodate at least 5000 samples distributed across multiple regions in California between 2016 and 2024.

Finalizing the Features to be Included:

Based on the scientific literature and preliminary exploratory data analysis, a robust set of predictive variables was finalized. These included:

- **NDVI**, computed from Sentinel-2 bands B8 (NIR) and B4 (Red), as a proxy for vegetation health and fire susceptibility;
- **Burn date**, **observed fire flag**, and **confidence level** extracted from FireCCI51;
- **Mean 2-meter air temperature**, along with the **u- and v- components of wind** from ERA5-Land, to account for meteorological drivers of fire spread.

  Each sample in the dataset encapsulated a time-stamped observation at a specific geographic point, ensuring that the resulting data matrix could be fed directly into a supervised regression model with minimal post-processing. The **confidence level** served as the continuous target variable, facilitating the use of models such as Random Forest Regressors, Gradient Boosted Trees, or Neural Networks depending on experimental design.

Development:

The development phase focused on implementing a fully automated data pipeline that seamlessly integrates geospatial extraction, feature engineering, and structured export. Using GEE's batch processing capabilities, scripts were developed to iterate over fixed geographic points in California, extract relevant variables across multiple years, and organize them into a tabular format. Each record captured the environmental and fire-related context for a specific date and location. The data was then pushed to Google Drive in CSV format using Earth Engine's Export table to drive functionality. Rigorous checks were incorporated to ensure data integrity, including filtering for missing values, handling cloud coverage in NDVI calculations, and validating temporal alignment across datasets. The pipeline was also modularized, allowing researchers to tune spatial resolution, temporal frequency, and sampling strategies without refactoring core logic.

Testing:

Once the data generation pipeline was operational, extensive testing was conducted to validate the system's reliability and accuracy. This included both unit testing of feature extraction routines and end-to-end validation using sample exports. Data was inspected for consistency in coordinate alignment, completeness of all required features, and logical coherence (e.g., fire observations should co-occur with elevated confidence values). Additionally, synthetic test coordinates were introduced to simulate edge cases such as regions with no fires or erratic weather patterns and water bodies. The regression targets were statistically profiled to ensure a balanced distribution suitable for learning, and Pearson correlation coefficients were computed to assess initial feature importance and guide future model tuning.

Evaluation:

The evaluation phase centered on assessing both the integrity of the data pipeline and the predictive capacity of the constructed regression dataset. Key indicators of pipeline robustness included the volume of successfully retrieved samples, the geographic and temporal coverage across California (2016–2024), and the consistency of feature extraction from satellite and meteorological sources.

To validate the formulation and usability of the dataset, baseline regression models—such as linear regression and decision trees—were trained on subsets of the data. These initial experiments aimed to quantify the extent to which the selected features (e.g., NDVI, temperature, and wind vectors) could explain the variation in wildfire detection confidence levels.

The evaluation phase focused on both the robustness of the data pipeline and the predictive strength of the regression formulation. Baseline models such as linear regression and decision trees were trained to assess how well environmental features could predict wildfire detection confidence. Standard metrics—**MAE**, **MSE**, **RMSE**, and **R^2**-were used to quantify model performance, revealing that meaningful patterns existed between variables like NDVI, temperature, wind components, and confidence levels. These early results confirmed the scientific validity of the problem setup and demonstrated that the system is scalable, reproducible, and well-positioned for integration into future wildfire early-warning systems.

3.4 Details of Hardware & Software

System Configuration:

Component	**Specification**
Processor	Intel Core i5 (12th Gen) 2.6 GHz
RAM	8 GB DDR4
Operating System	Windows 10 Pro, 64-bit

Software Used:

Software / Tool	**Version**	**Purpose / Usage**
Node.js	v14.17.3	Runtime environment to build and run the backend server
Express.js	v4.17.1	Lightweight web framework for creating and managing API endpoints
Google Earth Engine (GEE)	—	Core geospatial platform for satellite imagery and climate datasets (e.g. FireCCI, ERA5, Sentinel-2)
node-fetch	v2.6.1	HTTP client to interact with external services (e.g. OpenWeather, Nominatim)
CORS	v2.8.5	Middleware to enable cross-origin requests
body-parser	v1.19.0	Middleware for parsing incoming JSON request payloads
Visual Studio Code	—	Primary IDE for development, debugging, and code organization
Git	—	Version control and collaborative codebase management
TensorFlow	v2.9	For building and training regression models for wildfire prediction
NumPy	v1.21.3	Numerical computing library for array manipulation and mathematical operations

Pandas	v1.3.2	Structured data handling, preprocessing, and feature engineering
Matplotlib	v3.4.1	Data visualization: trends in satellite data and model predictions

3.5 Experimental Setup and Results

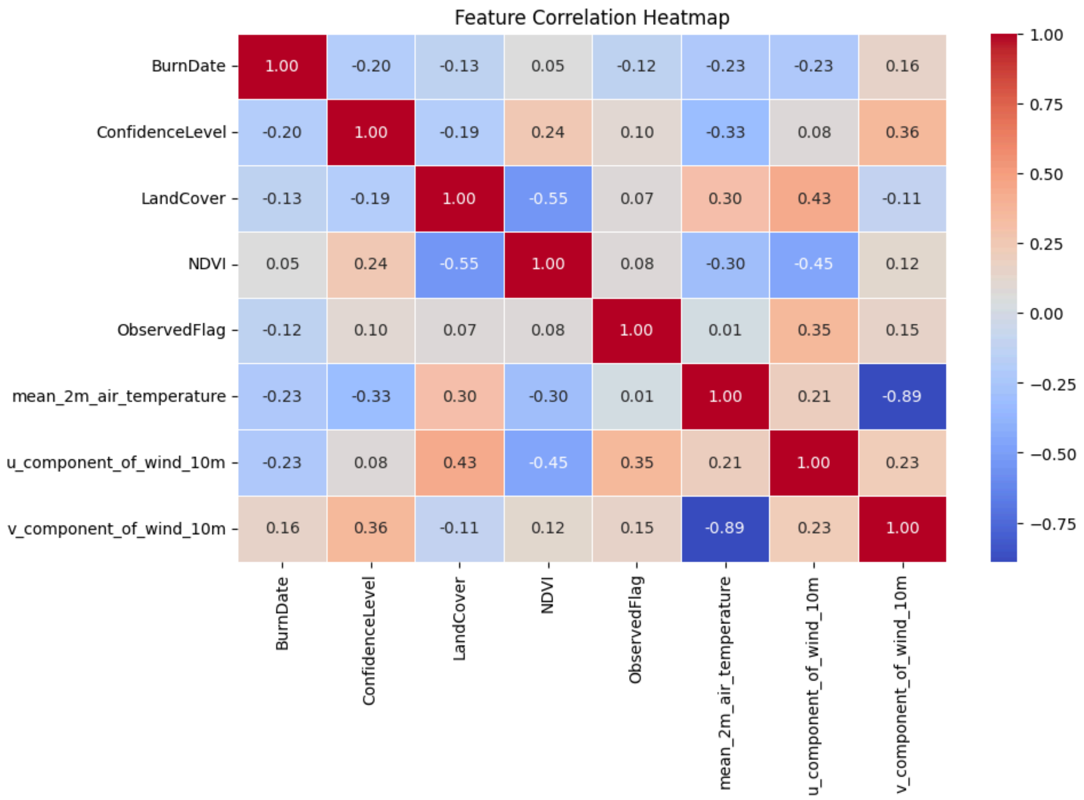

Image 3.5.1

A Pearson correlation heatmap (Figure 3.5.1) was generated to examine linear relationships among wildfire-related features. Most feature pairs exhibited weak to moderate correlations, suggesting limited multicollinearity.The main findings are summarized as follows:

Feature(s)	**Correlation**	**Interpretation**
BurnDate with other features	Weak	Burn occurrence data is largely independent of other features.
ConfidenceLevel with NDVI and v_component_of_wind_10m	≈ 0.24 and ≈ 0.36	Vegetation health and wind influence detection confidence.
LandCover and NDV	≈ -0.55	Certain land types have lower vegetation indices.
mean_2m_air_temperature and v_component_of_wind_10m	≈ -0.89	Temperature and vertical wind interact inversely.
u_component_of_wind_10m and v_component_of_wind_10m	≈ 0.23	Coupling between horizontal and vertical wind components
ObservedFlag with other features	Very Weak	Provides largely independent wildfire observation information.

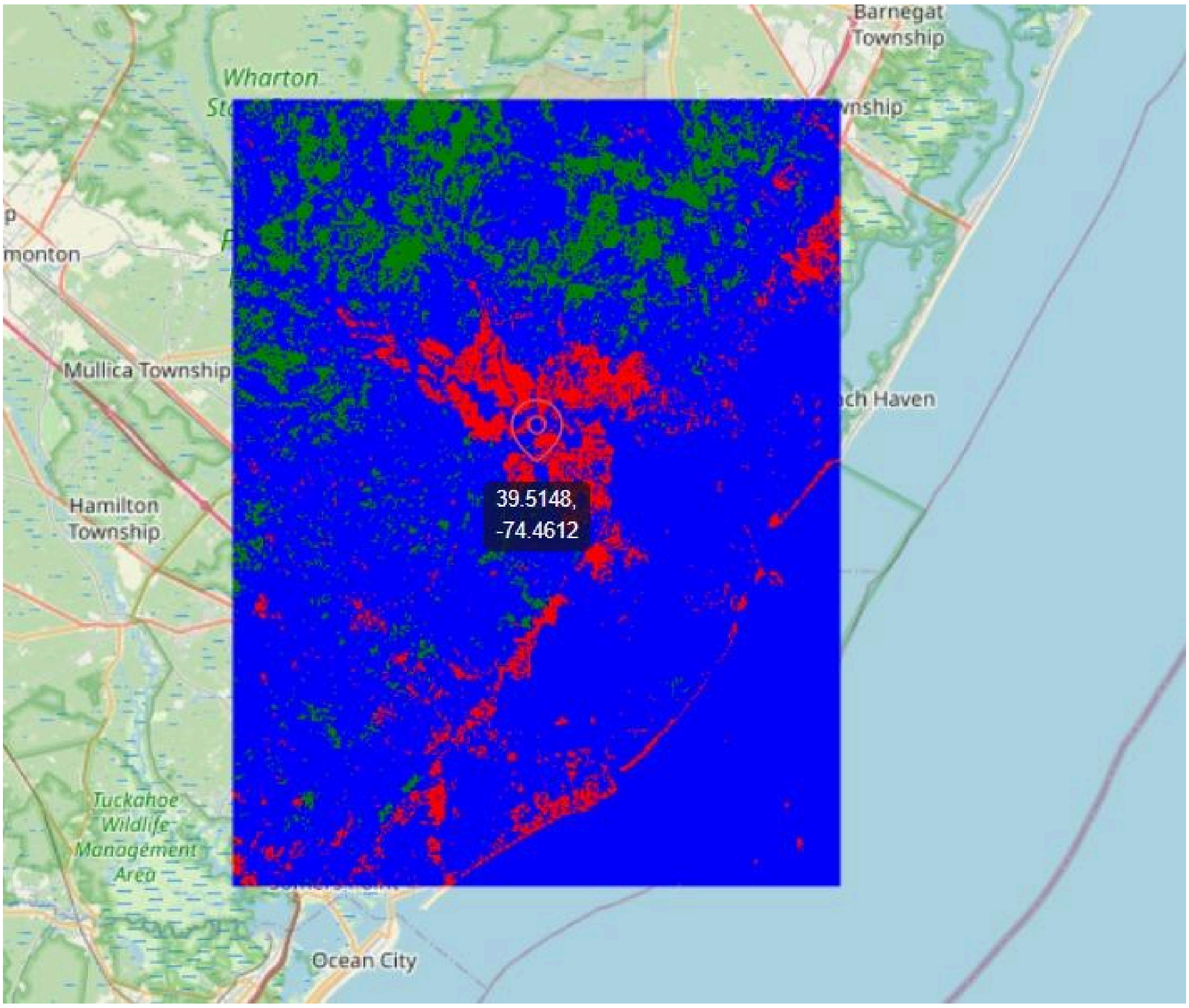

Image 3.5.2

The **K-Means clustering output** visualizes the segmentation of a geographic region based on NDVI and land cover characteristics, with seven distinct clusters identified. This unsupervised analysis aids in highlighting ecologically homogeneous zones that may respond similarly to fire hazards. The resulting visualization, overlaid on an interactive map interface, offers spatial insights that support localized model tuning and adaptive risk assessment across varied landscape types.

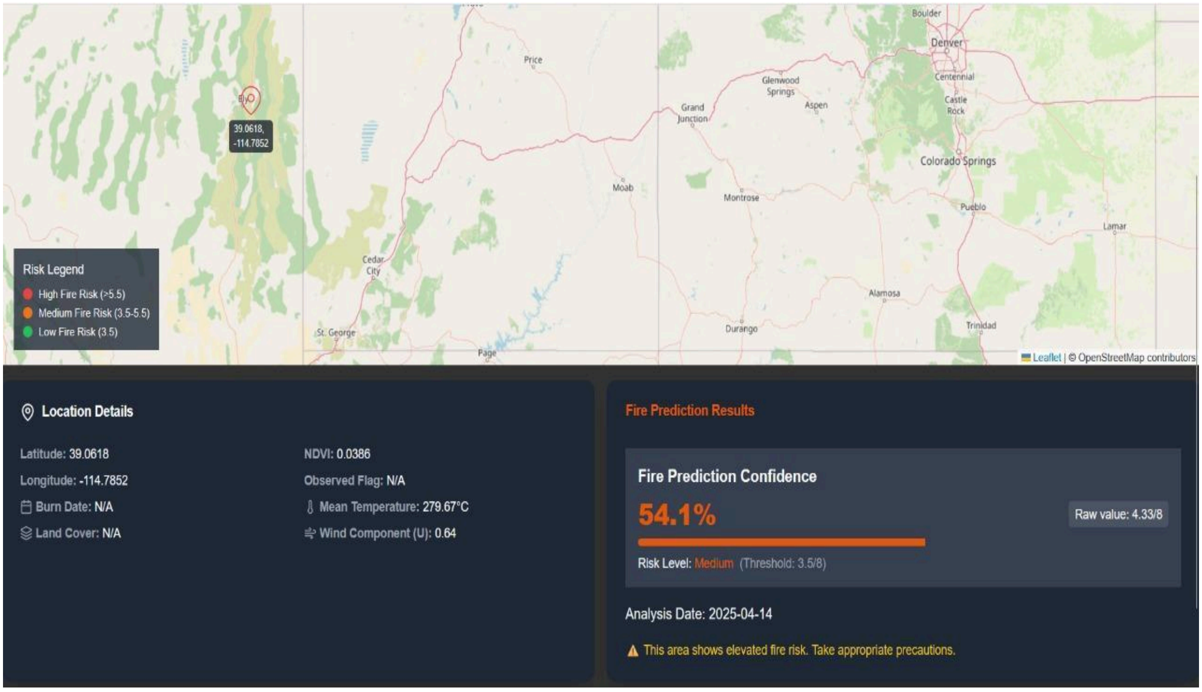

Image 3.5.3

The **fire prediction interface** presents real-time fire confidence outputs for user-selected coordinates. It displays relevant environmental parameters such as NDVI, temperature, and wind speed, alongside the model-generated fire confidence score (e.g., 54.1%) and an associated risk level (e.g., Medium). This interface converts raw satellite and meteorological data into actionable insights, intended for early warning and preparedness decision-making. The risk categorization and threshold-based alerts enhance the system's utility in field operations.

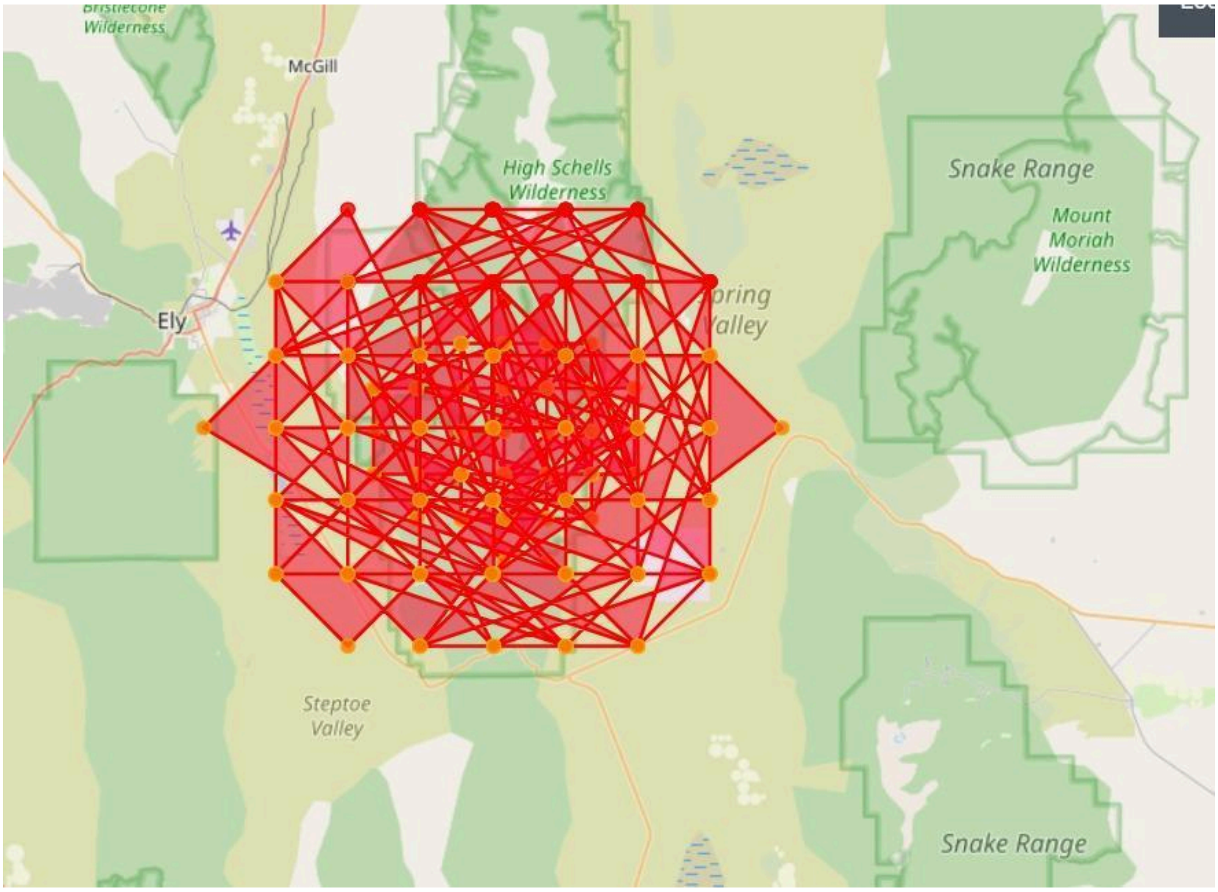

Image 3.5.4

The **fire spread visualization** depicts the spatial distribution of fire confidence estimates computed for a 5 km radius around a central location. Each node represents a prediction point, and the connecting edges signify potential spread directions. This high-density spatial mapping allows for anticipating fire propagation patterns and identifying high-risk zones. It serves as a foundational component for constructing geospatial fire spread models and supports the development of localized intervention strategies.

The above images predicts the fire levels in an area.

3.6 Conclusion and Future Scope

In conclusion, the development of the wildfire monitoring system represents a significant advancement in the field of wildfire detection and management. By leveraging modern machine learning techniques and integrating real-time data processing, our system enhances the accuracy and speed of wildfire detection, providing users with timely alerts and comprehensive information. The user-friendly interface and community engagement features further empower individuals to take proactive measures in their respective areas. As wildfires continue to pose serious threats globally, our project aims to contribute meaningfully to disaster preparedness and response efforts, ultimately safeguarding both natural ecosystems and human communities.

Looking ahead, there are several avenues for future work to enhance the capabilities of the wildfire monitoring system. One potential improvement involves expanding the machine learning model to include additional data sources, such as social media feeds and local weather stations, to further refine predictions and ensure real-time updates. Additionally, developing mobile applications for both iOS and Android platforms can improve accessibility and engagement, particularly in regions vulnerable to wildfires. Future iterations of the system could also incorporate advanced visualization techniques, such as augmented reality (AR), to provide users with immersive experiences when assessing fire risks. Lastly, continuous feedback from users will be essential in identifying areas for improvement, ensuring the system evolves to meet the needs of its users effectively

References

https://www.wildfiretracker.com

https://fires.globalforestwatch.org/

https://firemapper.com

https://www.fire.ca.gov/app/

https://firms.modis.gsfc.nasa.gov/

Ager, A. A., Kline, J. D., & Montgomery, C. A. (2019). "A machine learning approach to fire occurrence modeling in the western United States." *Forest Ecology and Management*, 453, 117661. DOI: 10.1016/j.foreco.2019.117661

Liu, Y., & Liu, J. (2021). "Real-time monitoring and prediction of wildfires using satellite data and machine learning." *Remote Sensing*, 13(1), 55. DOI: 10.3390/rs13010055

McCaffrey, S., & Olsen, C. (2012). "Social science to improve fuels management: A review of the literature." *Journal of Forestry*, 110(4), 203-207. DOI: 10.5849/jof.10-007

Balch, J. K., & Bradley, B. A. (2016). "Fire and Climate Change: The Need for a New Paradigm." *Ecology and Society*, 21(3). DOI: 10.5751/ES-08469-210330

Shakya, S., & Zhang, D. (2020). "Advanced data processing techniques for wildfire detection using satellite imagery." *International Journal of Remote Sensing*, 41(3), 831-850. DOI: 10.1080/01431161.2019.1679898