A

Mini Project Report

on

SmartFlood Mumbai: AI Risk Mapping using Bayesian Network

Submitted in partial fulfillment of the requirements for the

degree

Third Year Engineering – Computer Science and Engineering (Data Science)

by

ADITYA KATE	**23107126**
SUMAN MANIK	**23107056**
TANMAY HARMALKAR	**23107099**

Under the guidance of

Prof. Rajashri Chaudhari

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING (DATA SCIENCE)

A.P. SHAH INSTITUTE OF TECHNOLOGY

G.B. Road, Kasarvadavali, Thane (W)-400615

UNIVERSITY OF MUMBAI

Academic year: 2025-26

CERTIFICATE

This to certify that the Mini Project report on **SmartFlood Mumbai : AI Risk Mapping using Bayesian Network** has been submitted by **Aditya Kate (23107126)**, **Suman Manik (23107056)**, **Tanmay Harmalkar (23107099)** who are bonafide students of A. P. Shah Institute of Technology, Thane as a partial fulfillment of the requirement for the degree in **Computer Science Engineering (Data Science)**, during the academic year **2025-2026** in the satisfactory manner as per the curriculum laid down by University of Mumbai.

Ms. Rajashri Chaudhari
Guide

Dr. Pravin Adivarekar
HOD, CSE(Data Science)

Dr. Uttam D. Kolekar
Principal

External Examiner:
1.

Internal Examiner:
1.

Place: A. P. Shah Institute of Technology, Thane
Date:

ACKNOWLEDGEMENT

This project would not have come to fruition without the invaluable help of our guide **Ms. Rajashri Chaudhari** Expressing gratitude towards our HoD, **Dr. Pravin Adivarekar**, and the Department of Computer Science Engineering (Data Science) for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our Project coordinator **Ms. Aishwarya Londhe** and **Ms. Sarala Mary** for their valuable suggestions and ideas when we needed them. Additionally, we extend our thanks to our peers for their helpful suggestions.

TABLE OF CONTENTS

ABSTRACT

SmartFlood Mumbai, a Bayesian AI-based real-time flood prediction system designed for Mumbai's complex urban infrastructure. The system integrates Random Forest classification, K-means clustering, and Bayesian network inference to generate ward-level probabilistic flood risk assessments across 24 administrative divisions. Historical flood datasets comprising over one million records were combined with live meteorological inputs from the OpenWeatherMap API to enable adaptive, data-driven predictions. The architecture employs a microservices framework built on FastAPI fbackend computation and a React-based interface for visualization. A Leaflet-enabled web dashboard displays spatially resolved flood probabilities with color-coded ward representations. Furthermore, a flood-aware routing module applies A* pathfinding over OSMnx road networks, dynamically adjusting edge weights according to real-time flood conditions to identify safer navigation paths during rainfall events. The proposed system demonstrates that integrating probabilistic reasoning with machine learning improves both predictive accuracy and interpretability in urban flood modeling. The modular design facilitates extension to other flood-prone regions and supports integration with disaster management infrastructures for enhanced urban resilience.

Keywords: *Flood Prediction, Machine Learning, Bayesian Networks, Real-time Systems, Urban Planning, Mumbai, Interactive Mapping, Flood-aware Routing*

Chapter 1

Introduction

Flooding is one of the most common and destructive natural disasters affecting urban areas, posing a significant threat to human lives, infrastructure, and economic stability. This challenge is particularly acute in densely populated coastal cities like Mumbai, where annual monsoon rains bring extremely high precipitation within short time spans. Factors such as inadequate drainage capacity, encroachments on natural water channels, high tide levels, and rapid urbanization further aggravate the situation, leading to widespread waterlogging and severe flooding across multiple wards. These floods disrupt transportation networks, hinder daily livelihoods, damage public and private property, and in extreme circumstances, result in loss of life. Traditional meteorological systems, while effective at a macro level, typically provide forecasts on a city-wide or regional scale, failing to capture the highly localized nature of urban flooding. As a result, both citizens and disaster management authorities often lack timely, ward-specific insights required to take effective action.

To address this critical gap, the SmartFlood Mumbai: AI Risk Mapping project has been conceptualized as an intelligent, real-time flood prediction and management solution. By combining machine learning with live rainfall and water-level data, the system generates ward-level risk predictions that are significantly more actionable than conventional alerts. The architecture integrates multiple algorithms for comprehensive analysis: Random Forest for flood risk classification using historical and live data, K-Means clustering for identifying patterns and high-risk spatial zones, Bayesian Networks for modeling probabilistic dependencies and uncertainties among environmental factors, and the A* algorithm for suggesting safe evacuation routes during emergencies. The processed results are delivered through an interactive web platform that visualizes ward-specific risk zones, evacuation pathways, temporal trend charts, and comparisons with historical flood events. This system not only empowers citizens with localized, real-time intelligence for safer mobility and preparedness but also equips emergency response teams with actionable data for rapid interventions, while supporting urban planners in designing long-term strategies for climate resilience and disaster management.

1.1 Purpose

The purpose of this project extends beyond prediction to building a holistic framework for flood preparedness, response, and resilience in Mumbai. By delivering ward-level insights, the system ensures that the highly localized nature of flooding is captured, which is often overlooked in generic meteorological forecasts. This level of precision can significantly reduce disruption by enabling commuters to avoid flooded routes, residents to take precautionary measures in advance, and local authorities to issue targeted warnings rather than generalized alerts. Emergency response teams can prioritize deployment to wards with the highest predicted risks, thereby optimizing the use of limited resources and reducing response times. For policymakers and urban planners, the system offers valuable data-driven evidence to support long-term decisions related to infrastructure investment, drainage upgrades, and climate adaptation strategies.

On an academic and technological front, the project demonstrates the practical applicability of AI and data science techniques in addressing a pressing urban challenge. Random Forest enables accurate classification of flood risks across wards by learning from historical and live environmental data. K-Means clustering helps uncover hidden patterns, such as recurring high-risk zones or temporal trends across multiple monsoon seasons. Bayesian Networks address the inherent uncertainty in weather and flooding phenomena by modeling probabilistic relationships between factors like rainfall intensity, tide levels, and drainage capacity. The inclusion of the A* algorithm ensures that the system not only predicts risks but also guides citizens toward safe, optimized evacuation routes, bridging the gap between analysis and real-world action.

Furthermore, the project is designed with scalability and adaptability in mind. While it is tailored to Mumbai's unique geography and monsoon challenges, the modular design allows it to be adapted for other flood-prone urban regions. Its integration of real-time data streams, machine learning models, and geospatial visualization tools highlights how modern technology can transform disaster management into a proactive, intelligent, and community-focused system. By uniting prediction, planning, and response, SmartFlood Mumbai contributes not only to immediate safety but also to building long-term climate resilience for one of India's most vital metropolitan regions.

1.2 Problem Statement

Mumbai has historically endured the devastating effects of monsoon-induced flooding, with landmark events in 2005 and 2017 serving as reminders of its vulnerability. The 2005 deluge, one of the worst in the city's history, claimed over 1,000 lives, caused extensive damage to infrastructure, and left millions stranded without access to food, water, or transport for days. In 2017, a similar crisis unfolded, as heavy downpours once again crippled transportation systems, inundated homes, and highlighted how little progress had been made in strengthening flood preparedness. These events underscored systemic shortcomings, including the city's aging and inadequate drainage infrastructure, encroachments on natural water channels, and the lack of predictive systems that can anticipate ward-specific risks in real time.

1. **Limitations of Current Systems**

   Present-day meteorological services provide only broad, city-wide alerts, but these fail to capture localized flood vulnerabilities. Factors such as elevation differences, varying drainage capacities, proximity to the coastline, and unplanned urbanization create ward-level risks that remain unaddressed. Citizens, lacking hyper-local risk information, are often forced into dangerous conditions during daily commutes or routine activities.

2. **Challenges for Authorities**

   Municipal authorities and disaster management teams are unable to access high-resolution, ward-level flood predictions. This gap forces agencies into reactive mode, responding to crises only after they unfold rather than proactively mitigating risks. The delayed deployment of resources like rescue boats, medical support, and relief supplies increases casualties and economic losses. Moreover, the absence of integrated evacuation route guidance leaves both responders and citizens vulnerable when navigating floodwaters.

3. **Need for Advanced Solutions**

   There is a pressing need for an intelligent, ward-level flood risk management system that provides more accurate and actionable insights. Such a system should integrate real-time rainfall data and water-level monitoring with AI-based predictive models. The platform should empower citizens with timely, location-specific alerts, help emergency responders prioritize resource allocation, and support city planners in building long-term resilience strategies.

1.3 Objectives

The main objective of this project is to design and implement a ward-level, real-time flood risk prediction and management system for Mumbai, tailored to the city's recurring challenges during the monsoon season. To achieve this overarching goal, the system is guided by a set of focused and interlinked objectives.

1. Predict Flood Risk Using Random Forest

The system's first objective is to predict flood risk levels for each ward by using a Random Forest classifier. This model is trained on real-time rainfall intensity combined with additional factors such as tide levels and historical flood records. It classifies wards into categories like low, medium, high, and critical risk, providing precise and localized insights to help both citizens and authorities make informed decisions.

2. Identify Flood-Prone Zones with KMeans

The second objective is to group wards into flood-prone zones using KMeans clustering. By analyzing rainfall patterns across multiple wards, this technique reveals spatial relationships and identifies clusters of areas with higher vulnerability. This information supports better immediate response planning and informs long-term mitigation efforts.

3. Estimate Flood Probability with Bayesian Networks

The third objective uses Bayesian Networks to estimate flood probabilities under uncertainty. By modeling probabilistic dependencies between rainfall, tides, and drainage conditions, this approach maintains reliable predictions even when data is incomplete, variable, or rapidly changing, enhancing the robustness of the system's output.

4. Suggest Safe Evacuation Routes using A* Algorithm

The fourth objective employs the A* search algorithm to suggest safe travel or evacuation routes during flood events. By incorporating flood risk classifications into route planning, the system avoids high-risk wards and recommends paths that balance safety and efficiency, aiding both citizens and emergency responders.

5. Provide Data Through an Interactive Interface

The Fifth objective is to present these predictions and insights through an interactive user interface. This includes ward-wise maps with color-coded risk levels, trend charts, and dynamic dashboards.

1.4 Scope

The scope of this project encompasses the design, development, implementation, and testing of a comprehensive, AI-driven flood prediction and management system specifically tailored to Mumbai's urban and geographical context. The project focuses on providing real-time, ward-level intelligence that addresses the limitations of existing city-wide alert systems.

1. **Data integration and monitoring**

   The system integrates multiple data sources to maintain robustness and reliability. Historical rainfall data from opendatacity.in forms the foundation for training machine learning models, while real-time rainfall updates are fetched through the OpenWeatherMap API. Geospatial datasets such as ward boundaries provide the basis for mapping and visualizing flood risks. All datasets undergo preprocessing for accuracy and consistency to support predictive modeling and visualization.

2. **Stakeholder Benefits**

   The project broadly addresses the needs of several stakeholders. Citizens receive timely, ward-specific flood risk alerts to aid safe commuting and personal safety decisions. Emergency responders gain real-time insights into vulnerable areas for more effective resource deployment of rescue teams, relief materials, and medical assistance. City planners and policymakers benefit from long-term analytics to identify flood-prone areas and optimize drainage infrastructure and disaster preparedness strategies.

3. **Limitations and Dependencies**

   The accuracy and reliability of the system depend on the granularity, quality, and consistency of input data, particularly for rainfall and tide levels. Disruptions in real-time API feeds or issues within historical datasets may affect model performance. Additionally, extreme and unpredictable weather phenomena may fall outside historical data patterns, limiting prediction certainty.

   The real-time processing capability requires stable internet connectivity and server uptime, meaning accessibility could be impacted in severe disaster scenarios without backup solutions.

4. **Scalability and Future Adaptation**

Although currently limited to Mumbai, the system’s architecture is designed for scalability and adaptability. Its modular data pipelines, machine learning components, and visualization tools allow straightforward extension to other flood-prone urban areas, provided suitable datasets are present. Thus, this project not only addresses Mumbai’s immediate flood challenges but also establishes a replicable framework to bolster flood resilience in other metropolitan regions across India and beyond.

Chapter 2

Literature Review

This literature review systematically examines the existing research on urban flooding and flood risk management in Mumbai. The selected studies represent a broad spectrum of approaches, including hydrological modeling, geospatial techniques, social vulnerability assessment, technological innovations, and ecosystem-based solutions. By reviewing these contributions, the section highlights current understandings, significant findings, and ongoing challenges in Mumbai's flood governance. The aim is to present a critical synthesis of methodologies, results, and gaps to guide future research and policy development in urban flood resilience.

Tripathy et al. (2024) [1] conducted a combined analysis integrating traditional rainfall data and crowdsourced flood event data to study Mumbai floods. The methodology involved using parametric correction methods aimed at improving the accuracy of flood event detection. Despite efforts to enhance model reliability, additive and multiplicative corrections showed limitations, especially during periods of heavy rainfall when the corrections became ineffective. This demonstrates the challenges in flood modeling under extreme conditions and highlights the importance of integrating multiple data sources for effective flood monitoring.

Yash et al. (2024) [2] employed an integrated GIS and Analytic Hierarchy Process (AHP) approach to spatially delineate flood hazard zones in Mumbai. Their analysis incorporated various environmental parameters such as land use, elevation, drainage density, NDVI, and proximity to rivers and roads, combined with rainfall data. The multi-criteria decision-making framework allowed for nuanced flood risk mapping tailored to urban landscapes. However, the study acknowledged limitations related to the static nature of AHP and GIS methods and the difficulties in generalizing findings across dynamic urban contexts experiencing rapid development and changing flood dynamics. These standard limitations suggest the need for adaptive and real-time approaches.

Pathak et al. (2020) [3] adopted a multi-criteria approach combining social, economic, and environmental data to assess flood vulnerability at a sub-catchment level within Mumbai City. By integrating diverse datasets, the study aimed to capture the complexity of urban flood risks considering both physical exposure and social vulnerability. However, the complex integration of heterogeneous data posed challenges, particularly with variability in data quality and consistency.

Prajapat et al. (2025) [4] focused on multi-criteria flood risk assessment at a granular sub-city scale, evaluating exposures, sensitivities, and adaptive capacities. The empirical evidence from Mumbai highlighted how coarse urban land-use data limits detailed vulnerability assessments. Contextual issues at the sub-city scale, such as micro-topographic variation and socio-economic heterogeneity, require more refined data and localized study designs to accurately capture flood risk nuances. This highlights the trade-off between study scale and data resolution in flood vulnerability research.

Ali et al. (2024) [5] took a broader perspective by applying hydrological and hydrodynamic modeling (HOS=CaMa-Flood) to assess flood risk across large Indian river basins. The methodology considered dams, reservoirs, hazard intensity, exposure, and vulnerability factors to estimate flood risk comprehensively. Calibration of models was constrained by limited data availability, and uncertainties in exposure estimates introduced potential errors. This river-basin scale approach is critical for understanding flood dynamics at a macro scale but also demonstrates challenges in model accuracy due to data gaps and parameter uncertainties.

Chapter 3

Proposed System

The proposed system for SmartFlood Mumbai aims to revolutionize urban flood management by providing accurate, real-time risk predictions and navigation tools for Mumbai's 24 wards. Integrating machine learning with interactive visualizations and data processing, SmartFlood Mumbai empowers emergency responders, planners, and citizens to mitigate flood impacts through proactive assessments and safe routing.

1. The System Architecture: SmartFlood Mumbai follows a scalable four-layer design, including a React-based frontend for interactive dashboards, a FastAPI gateway for API management, machine learning services for predictions, and a data layer for historical and real-time inputs.

2. Frontend Interface: The web dashboard uses React and Leaflet to display color-coded ward maps, real-time weather data, and flood-aware routing, allowing users to access detailed predictions and plan safe paths via tabbed navigation.

3. API Gateway: Implemented with FastAPI and Uvicorn, this layer manages RESTful endpoints for predictions, weather retrieval, and routing, incorporating CORS, error handling, caching, and fallback mechanisms for reliability.

4. Machine Learning Models: An ensemble approach combines a Random Forest classifier for risk classification, K-means clustering for ward zoning, and Bayesian Networks for probabilistic uncertainty quantification, trained on over 1 million historical records [1] [4].

5. Data Integration: The system uses enriched datasets with weather parameters, GeoJSON boundaries, and real-time feeds from OpenWeatherMap for conditions, plus mock tide data and OSMnx for road networks [6] [7].

6. Routing and Optimization: A* algorithms with dynamic weights calculate safe routes avoiding high-risk areas, supported by multi-level caching for predictions and graphs to ensure fast, concurrent performance.

By implementing these features, SmartFlood Mumbai aims to provide a comprehensive platform for predictive flood management, enhancing safety and decision-making in urban environments

3.1 Features and Functionality

Real-Time Flood Risk Prediction:

The system provides instantaneous flood risk assessments for all 24 Mumbai wards using a combination of current weather data and historical patterns. Users can access individual ward predictions through an interactive map interface, with each ward displaying color-coded risk levels ranging from low (green) to high (red) based on the ensemble model predictions.

Interactive Ward Visualization:

The Leaflet-based mapping system enables users to explore flood risk data through an intuitive geographic interface. Each ward is clickable, providing detailed flood prediction information including risk probability, confidence levels, weather conditions, and historical flood frequency data. The visualization supports zoom, pan, and layer controls for detailed exploration of specific areas.

Flood-Aware Navigation System:

The routing component calculates optimal navigation paths that avoid high-risk flood areas using A* pathfinding algorithms with dynamic edge weights. Users can input start and end locations to receive safe route recommendations that balance travel time with flood risk avoidance, with blocked road segments clearly marked on the map interface.

Weather Integration Dashboard:

Real-time weather widgets display current conditions for selected wards including rainfall measurements, temperature, humidity, wind speed, and tide levels. The weather service integrates with OpenWeatherMap API while maintaining fallback capabilities for service reliability.

Batch Prediction Processing:

The system supports simultaneous flood risk assessment for all 24 Mumbai wards, enabling comprehensive city-wide flood risk monitoring. Batch processing includes summary statistics, high-risk ward identification, and city-wide risk distribution analysis.

Model Performance Monitoring:

Comprehensive health check endpoints provide real-time system status including model availability, prediction accuracy metrics, service connectivity, and performance statistics. The system maintains detailed logs of prediction requests, response times, and error rates for continuous improvement.

Ward Clustering Analysis:

The K-means clustering component groups Mumbai wards into four distinct risk zones based on historical flood patterns, elevation data, drainage capacity, and population density. This clustering enables targeted intervention strategies and resource allocation for high-risk areas.

Probabilistic Risk Assessment:

The Bayesian Network component provides uncertainty quantification for flood predictions, delivering confidence intervals and probabilistic risk assessments that account for the inherent uncertainty in weather prediction and urban flood dynamics.

Caching and Performance Optimization:

Multi-level caching strategies ensure sub-second response times for individual ward predictions while maintaining system scalability. The system implements intelligent cache invalidation and preloading strategies for optimal performance.

Fallback and Error Handling:

Comprehensive fallback mechanisms ensure system reliability during external service disruptions. The system maintains mock data generation capabilities and graceful degradation strategies that preserve core functionality even during weather API outages or network connectivity issue.

Chapter 4

Requirements Analysis

For the requirement analysis of the SmartFlood Mumbai: AI Risk Mapping project, we need to identify and document the functional and non-functional requirements that the system must meet to fulfill its objectives effectively. Here's a breakdown of the requirement analysis for SmartFlood Mumbai:

A. Functional Requirements:

1. **Data Collection and Integration**:

   The system should collect real-time rainfall and tide data from APIs such as OpenWeatherMap and integrate it with historical rainfall and flood datasets to ensure comprehensive data availability for predictions.

2. **Data Preprocessing:**

   The system should preprocess collected data to remove inconsistencies, missing values, and other errors, preparing it for accurate analysis and modeling.

3. **Flood Risk Prediction:**

   The system should analyze processed data using machine learning models like Random Forest to predict flood risk levels for each of Mumbai's 24 wards.

4. **Ward Risk Zoning:**

   The system should group wards into high-risk zones using KMeans clustering based on historical and environmental factors to enable targeted risk assessment.

5. **Probabilistic Likelihood Estimation:**

   The system should estimate probabilistic flood likelihood with Bayesian Networks, providing uncertainty quantification for more reliable predictions

6. **Safe Route Suggestion:**

   The system should suggest optimal safe evacuation or travel routes using the A* search algorithm, incorporating real-time flood risk data.

7. **Visualization and Dashboards:**

   The system should visualize results on an interactive map with color-coded risk levels, display trend charts and dashboards, and allow comparisons with historical flood events.

B. Non-Functional Requirements:

1. **Real-Time Performance:**

   The system should generate predictions and route suggestions quickly to ensure they are actionable during ongoing flood events.

2. **Accuracy:**

   The system should maintain high prediction accuracy to avoid misleading users and ensure safety in flood management decisions.

3. **Reliability and Robustness:**

   The system should handle large volumes of data and multiple simultaneous user requests without failures or crashes.

4. **Usability:**

   The system should feature an intuitive and interactive interface that allows non-technical users to easily understand and navigate flood risk information.

5. **Software Infrastructure:**

   The system should utilize Python for backend development, Scikit-learn for machine learning, FastAPI or Flask for API handling, and React.js with Leaflet.js and Chart.js for frontend visualization to support efficient processing and user experience

Chapter 5
Project Design

5.1. Use Case diagram

Use Case Diagram of SmartFlood Mumbai (Fig 5.1) captures the main functionalities and interactions between user and System. It shows how different users interact with a flood prediction system for Mumbai and what the system does internally.

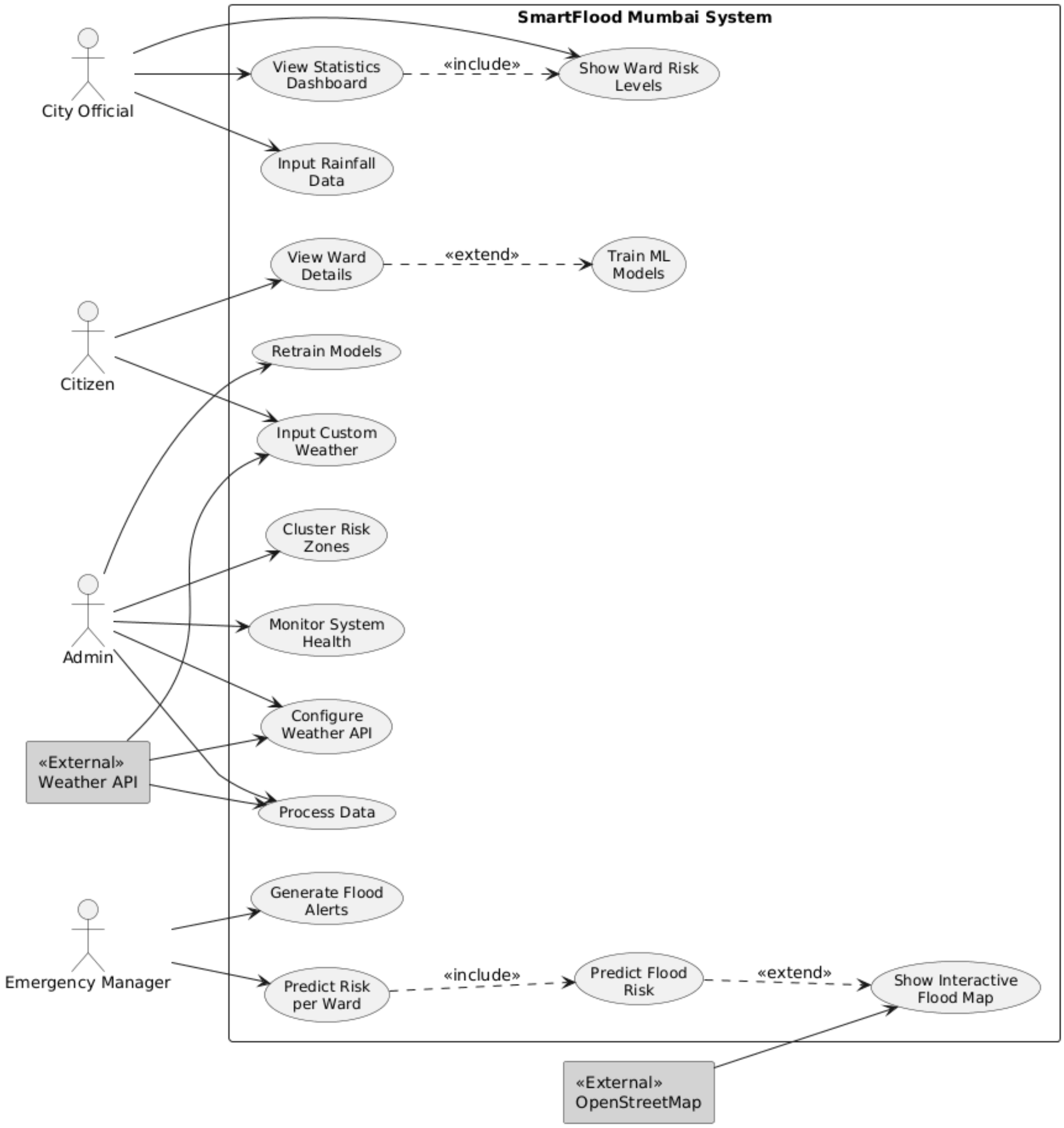

Fig 5.1. Use Case Diagram

The Figure 5.1 illustrates the use case diagram of the SmartFlood Mumbai system, representing the functional interactions among various users, subsystems, and external data sources that collectively enable real-time flood prediction and management across Mumbai's 24 administrative wards. The diagram defines how each actor contributes to or benefits from the system, outlining both the internal and external processes that form the foundation of the flood prediction and response mechanism.

The SmartFlood Mumbai system is built to serve multiple user categories involved in flood preparedness and disaster management. The primary actors include Citizen, City Official, Emergency Manager, and Admin, while two external entities - OpenWeather API and OpenStreetMap-supply live data streams essential for predictive modelling and geospatial visualization. Each actor performs distinct yet interconnected functions to ensure that the system operates efficiently and delivers accurate, real-time insights into urban flood risks.

A. System Overview and User Interactions

At its core, the system integrates Artificial Intelligence (AI) and Machine Learning (ML) models with probabilistic reasoning to predict ward-wise flood risks. The SmartFlood Mumbai system receives continuous data inputs from the OpenWeather API, which provides live weather parameters such as rainfall, humidity, and temperature. Simultaneously, OpenStreetMap contributes geospatial data, enabling the system to render interactive maps that display flood-prone areas and optimal routes under varying conditions.

1. Citizens interact with the system primarily for awareness and personal safety. They can input custom weather data to simulate potential scenarios and view ward-level flood details to assess the safety of their locality. The interactive flood map provides visual information on high-risk zones using color-coded layers, allowing residents to make informed decisions during rainfall events.
2. City Officials utilize the system for administrative monitoring and planning. Through the dashboard, they can observe ward-level flood risk levels, examine rainfall statistics, and access aggregated data visualizations. These features support evidence-based policymaking, enabling government authorities to allocate resources effectively and issue timely public advisories.

3. Emergency Managers serve as operational users who respond to real-time flood events. They rely on predictive insights to identify high-risk wards, generate flood alerts, and plan evacuation or relief activities. By accessing the system's monitoring and alerting features, Emergency Managers ensure rapid decision-making during emergencies, improving coordination among response teams.
4. Administrators (Admins) are technical users responsible for maintaining system performance and accuracy. Their functions include configuring the weather API, processing and cleaning data, and training or retraining machine learning models to enhance predictive accuracy. Admins also oversee system health, ensuring that the backend infrastructure and data pipelines operate without interruption.

B. Internal System Processes

The SmartFlood Mumbai system integrates several internal modules, each performing a specialized task. The Input Custom Weather module allows users to manually provide weather data for what-if simulations, testing how specific rainfall or tide values influence flood probabilities. The Process Data module handles data ingestion, preprocessing, and feature extraction, ensuring that inputs are structured correctly for model inference.

The Cluster Risk Zones component employs K-means clustering to group wards with similar flood characteristics. This clustering helps in categorizing areas into low, medium, and high-risk zones dynamically, based on both historical and real-time data. The Predict Risk per Ward and Predict Flood Risk modules use a hybrid model combining Random Forest classifiers and Bayesian Networks to compute flood probabilities for each administrative division. These models quantify uncertainty, ensuring that decision-makers understand both the likelihood and confidence of predictions.

The Show Interactive Flood Map module integrates with OpenStreetMap data to display real-time flood levels and risk intensities. It also supports A* pathfinding for flood-aware routing, where the system dynamically adjusts route weights based on flooded road segments, enabling safe navigation during extreme events. The Generate Flood Alerts module extends this predictive capacity by issuing warnings when risk thresholds are exceeded.

C. Actor Roles and Responsibilities

Each actor's role aligns with specific use cases represented in the diagram:

1. **Citizen** - Engages primarily with visualization and self-assessment tools. Key use cases include "Input Custom Weather," "View Ward Details," and "Show Interactive Flood Map." These functions empower individuals to understand and respond proactively to flood risks in their vicinity.

2. **City Official** - Oversees administrative monitoring and policy implementation. Key interactions include "Show Ward Risk Levels," "View Statistics Dashboard," and "Monitor System Health." This supports strategic planning and long-term flood resilience measures.

3. **Emergency Manager** - Utilizes operational tools for real-time crisis response. Main interactions include "Generate Flood Alerts," "Predict Flood Risk," and "View Interactive Flood Map." These tools provide actionable intelligence for managing emergency response efforts.

4. **Admin** - Manages technical and operational maintenance of the system. Major activities include "Configure Weather API," "Retrain Models," "Train ML Models," and "Monitor System Health." Admins ensure model reliability and system uptime.

5. **External Systems:**

    - OpenWeather API provides live meteorological data such as rainfall, temperature, and humidity, crucial for continuous prediction updates.

    - OpenStreetMap contributes real-time geographic information for mapping, pathfinding, and visualization.

D. Relationships and Dependencies

The use case diagram employs both «include» and «extend» relationships to represent dependencies and conditional behaviors within the system.

- The «include» relationships signify that one function inherently requires another to execute. For example, Show Interactive Flood Map includes Predict Flood Risk and View Ward Details, as the map visualization depends on predicted risk data and ward-level information. Similarly, View Statistics Dashboard includes Show Ward Risk Levels, ensuring that statistical summaries always reflect the latest risk computations.
- The «extend» relationships indicate optional or conditional activities that enhance base functionalities. Generate Flood Alerts extends Predict Flood Risk, as alerts are only generated when risk predictions exceed defined safety thresholds. Likewise, Retrain Models extends Train ML Models, reflecting that retraining occurs periodically or upon detecting significant model drift or data updates.

These relationships collectively demonstrate the system's modular and dynamic architecture, where core processes can operate independently while enabling extensions for specialized scenarios.

E. System Significance

The SmartFlood Mumbai use case model emphasizes a scalable, data-driven architecture that connects prediction, visualization, and decision support in a single integrated framework. By involving diverse actors-ranging from citizens to administrators-the system bridges the gap between advanced AI analytics and public usability. The inclusion of external data sources ensures real-time adaptability, while the modular design enables future extension to other flood-prone regions or integration with national disaster management platforms.

Overall, Figure 5.1 encapsulates the logical flow of interactions, functional dependencies, and actor responsibilities within the SmartFlood Mumbai ecosystem. It highlights how data acquisition, predictive modeling, and visualization converge to create a proactive and resilient flood management system tailored for complex urban environments.

5.2. DFD (Data Flow Diagram)

This data flow diagram (fig 5.2.1 and fig 5.2.2) for SmartFlood Mumbai shows the logical, modular design for a robust flood prediction and alert system, optimizing for accurate forecasting and actionable, real-time guidance for Mumbai residents.

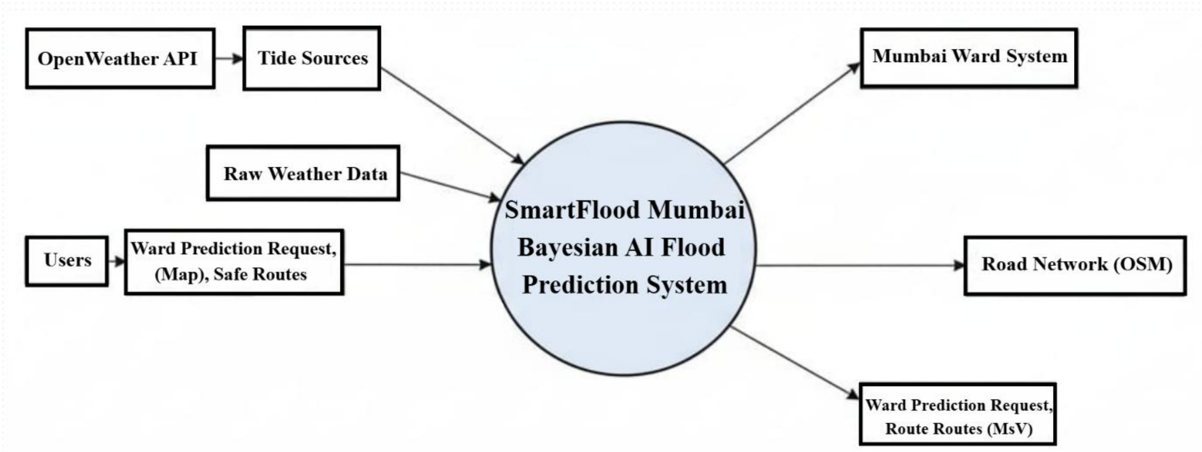

fig 5.2.1 Level 0 - Data Flow Diagram– SmartFlood Mumbai

The figure 5.2.1 Level 0 DFD provides an overall, high level view of the dataflows, showing the main external sources and output interactions.

The central process of the system is the SmartFlood Mumbai Bayesian AI Flood Prediction System, which functions as the core processor for all incoming and outgoing information. It receives input from several external entities, primarily the OpenWeather API and tide data sources, which provide real-time and historical weather and tide information. Users also interact directly with the system by submitting ward-level flood prediction requests, accessing interactive maps, and seeking safe travel routes during flood situations. The system processes these inputs by integrating them with internal data sources such as the Mumbai Ward System and the Road Network (OpenStreetMap), both of which are essential for generating accurate, localized predictions and for mapping the safest routes under dynamic flood conditions. The resulting outputs include ward-level flood prediction results and optimized safe routes, which are delivered to users and can also be utilized by external systems for route planning and disaster management purposes.

This level emphasizes data aggregation across weather, tide, ward, road, and user inputs, centralizing the information for comprehensive risk-aware flood prediction and decision support.

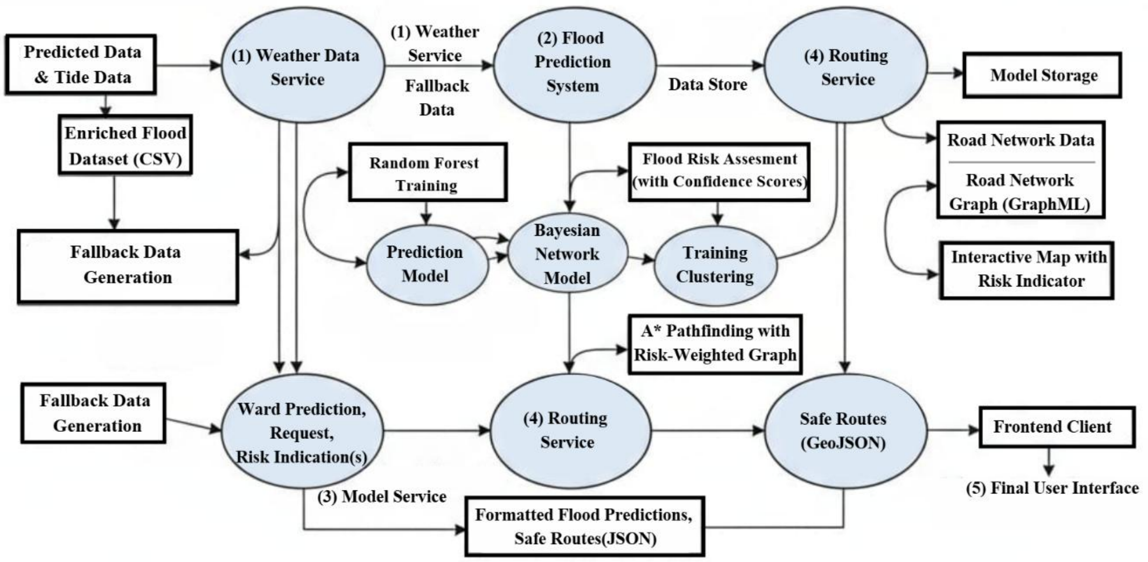

fig 5.2.2 Level 1 – Data Flow Diagram– SmartFlood Mumbai

The figure 5.2.2 Level 1 DFD decomposes the central process into four key components, illustrating how the data moves within the system, the sequence of computation, and data transformation at each step.

The system captures a feedback loop for model training and refinement, linking outputs with updated inputs, thus allowing dynamic re-prediction and improved routing as situations evolve. The clear flow ensures real-time, data-driven, confident flood warnings and safe route guidance for end users.

The diagram represents the DFD Level 1 of the SmartFlood Mumbai Bayesian AI Flood Prediction System. It outlines the logical sequence of processes, the movement of data, and the interaction between major functional modules within the system. Unlike a traditional architecture diagram that focuses on infrastructure, software components, and deployment layers, this diagram emphasizes how data is transformed and propagated through various intelligent subsystems to generate actionable flood predictions and safe travel routes in real time.

The system initiates with the Weather Data Service, which functions as the data ingestion and preprocessing unit. It continuously collects both live and forecasted weather and tide data from external APIs such as OpenWeather, integrating them with historical datasets to ensure temporal continuity. In case of data unavailability or sensor downtime, a Fallback Data Generation mechanism is triggered to synthesize approximate values using statistical interpolation and historical correlation models. This ensures no data gaps in the pipeline. The service then produces an Enriched Flood Dataset (CSV) - a standardized and quality-checked dataset ready for analytical use - which is passed downstream for model training and prediction.

Next, the Flood Prediction System operates as the analytical core of the pipeline. It begins by training a Random Forest model that performs deterministic flood prediction using structured features such as rainfall intensity, tide height, and ward-level drainage efficiency. The output from this model is further processed by a Bayesian Network Model, which integrates probabilistic reasoning to capture dependencies between environmental factors and to assess uncertainty in the predictions. The module performs Flood Risk Assessment by attaching confidence scores to each ward-level prediction, representing the likelihood of flooding under current and forecasted conditions. To enhance the model's adaptability, training and clustering processes are executed periodically to refine prediction accuracy and identify spatial flood behavior patterns among similar wards.

The third module, the Model Service, serves as the operational interface between the predictive models and external queries. It handles model deployment, allowing the system to provide real-time flood risk indications for individual wards. It manages prediction requests, response formatting, and the continuous retraining pipeline to ensure that the deployed models remain up to date with evolving weather patterns and tide variations.

Following this, the Routing Service utilizes the predictive outputs and transforms them into spatially meaningful insights. It ingests flood risk data, accesses the Road Network Graph (GraphML) built from OpenStreetMap (OSM) data, and constructs a risk-weighted road network graph. Using algorithms like A* (A-star) pathfinding, the service dynamically computes safe travel routes that balance minimal distance with minimal flood exposure. These routes are formatted into GeoJSON for geospatial compatibility and further integrated into the interactive flood map interface.

5.3. System Architecture

This architecture diagram (fig 5.3.) provides a systematic, modular overview of how the "SmartFlood Mumbai" system works from input to output, highlighting both data and model flow.

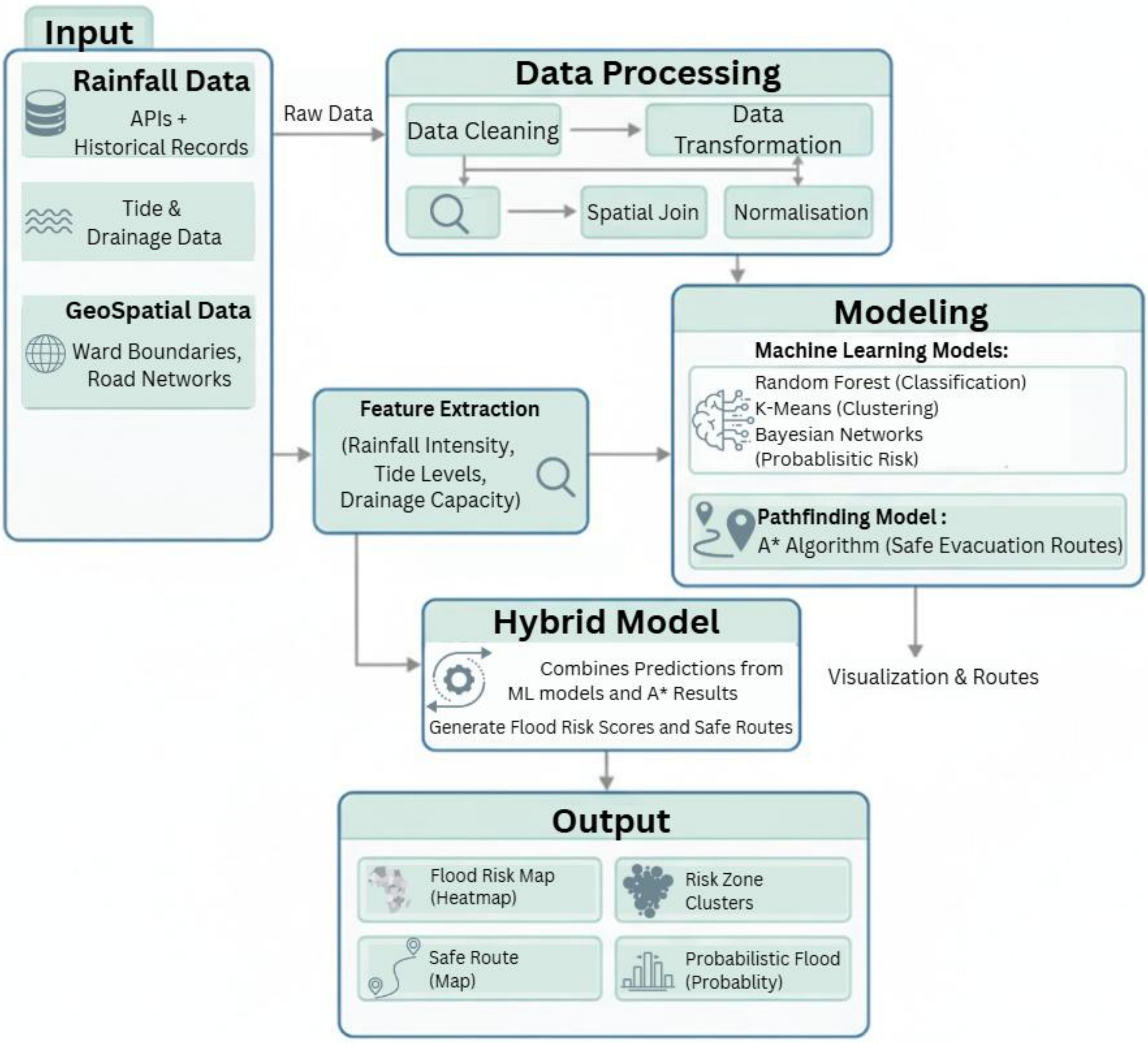

Fig 5.3: System Architecture

The figure 5.3. Architecture Diagram breaks down the platform into clear building blocks: Input, Data Processing, Modeling, Hybrid Model, and Output.

The system begins by gathering comprehensive input data from multiple sources, including rainfall information collected through APIs and historical records that capture both real-time measurements and archival patterns. This is supplemented by tide and drainage data, which are particularly crucial for flood forecasting in coastal Mumbai, along with geospatial data containing ward boundaries and detailed road network maps essential for accurate mapping and routing capabilities.

Once collected, this raw data enters a rigorous processing phase where it undergoes cleaning to eliminate noise and inconsistencies, followed by transformation to prepare it for analysis. A spatial join operation aligns geographic information such as ward boundaries and road segments with the flood and weather datasets, while normalization standardizes values across different data sources to ensure comparability and reliability in subsequent modeling stages.

The feature extraction phase then summarizes and computes critical variables from the cleaned data, including rainfall intensity, tide levels, and drainage capacity. These extracted features serve as fundamental inputs for both risk modeling and safe route calculation. The system employs multiple machine learning models in its modeling phase, utilizing Random Forest for risk classification, K-Means clustering to group wards or zones with similar risk profiles, and Bayesian Networks to predict probabilistic flood risk based on numerous interdependent factors. Alongside these, a pathfinding model implements the A* algorithm to identify optimal, safe evacuation routes across Mumbai's road network in real time.

These various analytical approaches converge in a hybrid model that combines predictions and scores from the machine learning components-such as risk zones and flood probabilities-with the A* algorithm's safe route calculations. This integration produces a fused output that balances accuracy with practical applicability. Finally, the system generates multiple outputs including a flood risk heatmap that visualizes predicted severity across the city, risk zone clusters highlighting groups of wards with similar significant flood risks, safe route maps showing users the safest available paths while avoiding flooded or impassable roads, and probabilistic flood assessments presenting calculated likelihood of flooding for each location.

5.4. Implementation

The Implementation diagram presented for the SmartFlood Mumbai system. Each diagram visualizes a different part of the deployed user interface and data-driven functionality of your flood prediction and routing project.

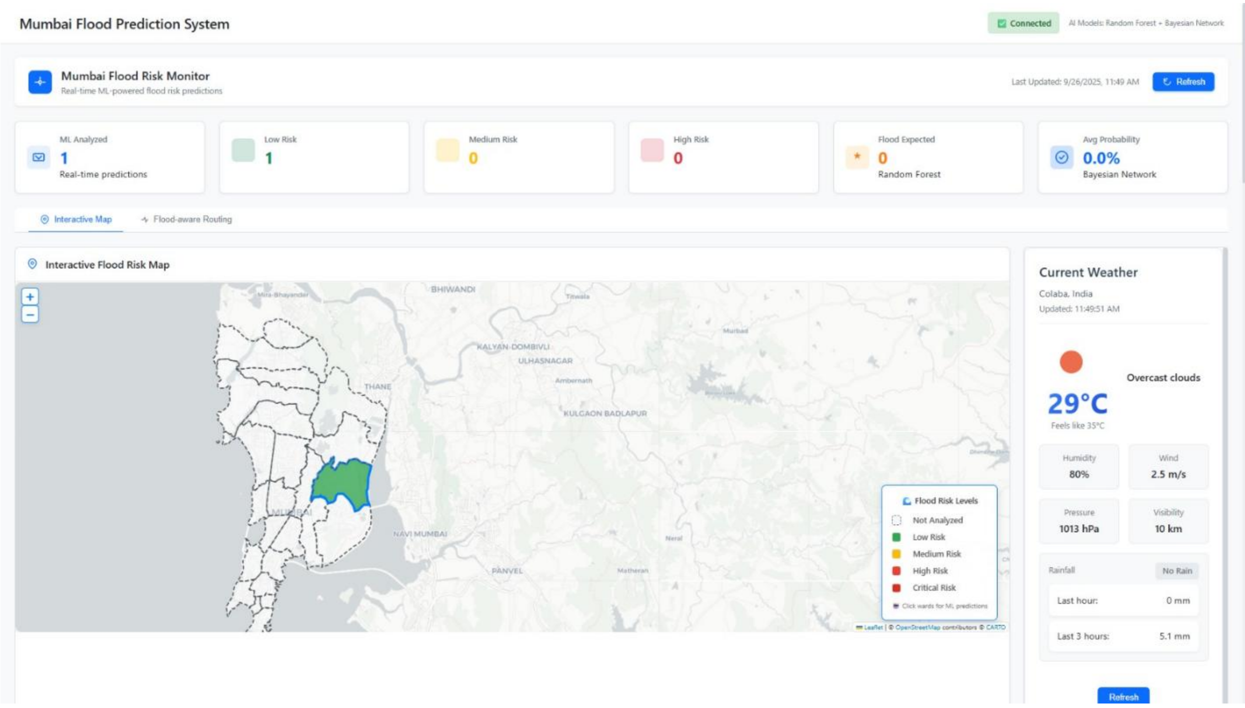

Fig 5.4.1: Mumbai Flood Prediction Dashboard Interface

The figure 5.4.1 depicts the Mumbai Flood Prediction Dashboard, which centrally displays real-time flood risk statistics including the count of currently analyzed locations, categorized flood risk levels, flood event expectations, and average probabilities derived from AI models such as Random Forest and Bayesian Network. It features an interactive map that visually highlights flood-affected areas across the city, alongside a flood-aware routing panel where users can input geographic coordinates or click on the map to set start and end points, with the system calculating and displaying safe routes that avoid flood-prone zones based on real-time risk assessments

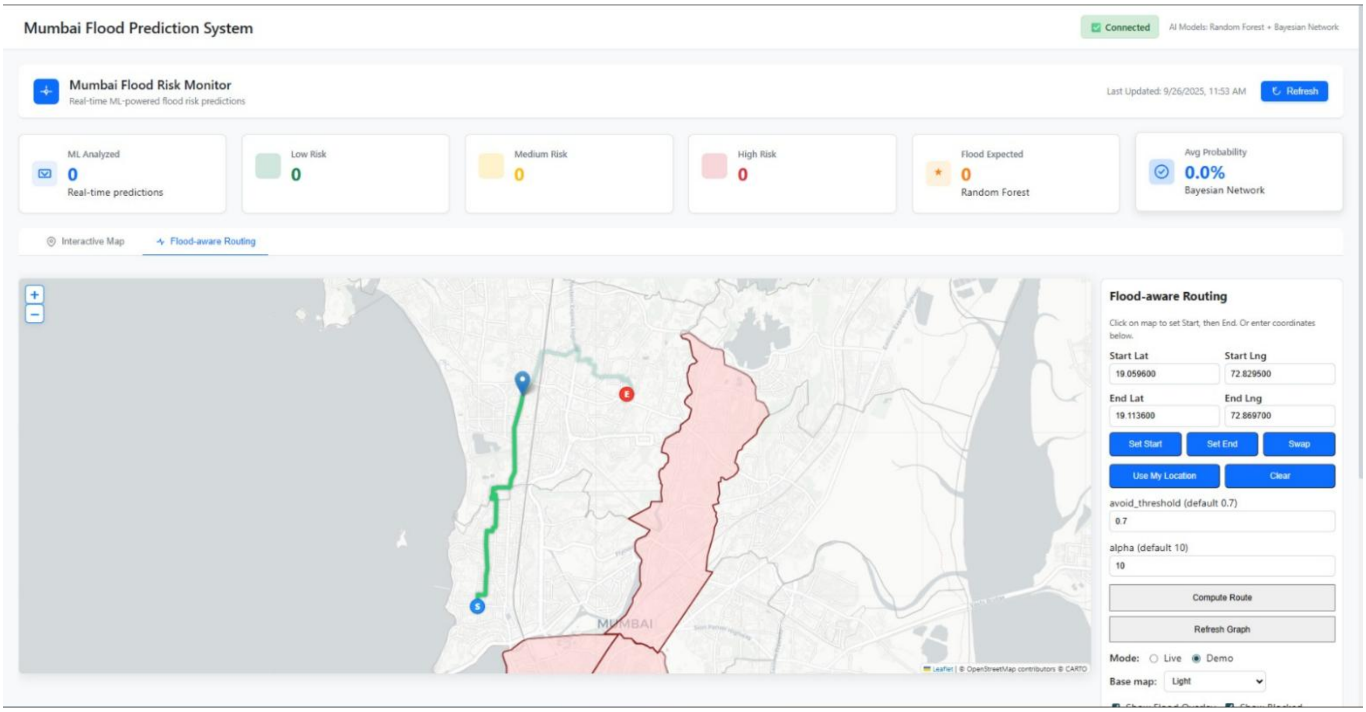

Fig 5.4.2: Mumbai Flood Prediction Routing Dashboard

The figure 5.4.2 shows On , the flood-aware routing form: users enter latitude/longitude, set start/end, adjust algorithm parameters (avoid_threshold, alpha), and compute a recommended route. The results reflect real flood data and risk maps.

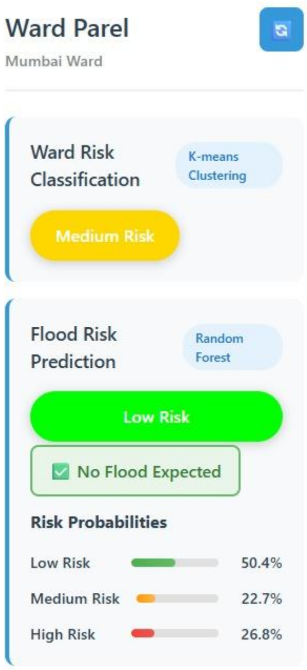

Fig 5.4.3: Ward-Level Risk Details and Probabilities

The figure 5.4.3 presents detailed ward-level flood risk data, exemplifying how specific localities such as the Parel ward are classified using clustering techniques like K-means and how Random Forest models contribute to overall flood risk predictions. This section combines categorical labels like "Medium Risk" or "Low Risk" with clear visual cues such as color-coded badges and risk probability bars for more granular understanding and transparency.

Fig 5.4.4: Live Flood Risk & Weather Side Panel

The figure 5.4.4 focuses on the flood risk and current weather status panel, showing an easy-to-read flood risk level indicator such as "Low Flood Risk," with corresponding flood probability percentages, while providing detailed environmental conditions like temperature, humidity, rain type (e.g., drizzle), wind speed, and a recent rainfall analysis summarizing the quantity of rain in the last hour and past three hours.

Chapter 6

Technical Specification

The technical specifications of a project outline the critical architectural elements, tools, and technologies that underpin its development and operation. This section provides a detailed description of the system architecture, data management approaches, algorithms employed, and the specific software and hardware components integrated into the project. It includes information about the frameworks, databases, APIs, and user interface technologies used, as well as details on security measures, accessibility considerations, and performance optimization strategies.

Technical specifications are essential for understanding how the project is built and maintained; offering insights into its scalability, reliability, and the user experience it delivers. This documentation is crucial for developers, stakeholders, and technical teams involved in the project as it serves as a blueprint for current operations and future enhancements. By detailing the technical environment and the interactions between various components, these specifications help ensure that the project meets its intended functionality and performance goals efficiently and effectively.

- **Frontend:**

  1. React.js

  2. Leaflet.js

  3. Chart.js

Technologies: The frontend of SmartFlood Mumbai is developed using React.js, which provides a dynamic and interactive user interface for real-time flood risk mapping and navigation.

Visualization Libraries: Leaflet.js is employed for displaying interactive maps with ward boundaries and risk levels, while Chart.js handles graphical representations of rainfall, water level data, and prediction trends.

- **Backend Development:**

  1. Python 3.10

  2. FastAPI or Flask

Libraries and Tools:

1. **API Data Handling**: NumPy and Pandas are used for data preprocessing and management, ensuring clean integration of historical and real-time datasets.

2. **Machine Learning**: Scikit-learn implements models such as Random Forest and KMeans clustering, while Pgmpy or PyMC3 supports Bayesian Network modeling for probabilistic estimates.

3. **Routing and Visualization**: NetworkX enables the A* algorithm for safe route planning, Matplotlib aids in data visualization, GeoJSON provides ward-level boundary mapping, and the OpenWeatherMap API delivers live rainfall and weather updates.

Algorithms:

1. Random Forest Classifier:

The Random Forest Classifier serves as the primary machine learning model for predicting flood risk levels across different geographical zones in Mumbai. This ensemble learning method operates by constructing multiple decision trees during the training phase and outputting the mode of the classes for classification tasks.

The algorithm analyzes historical flood data combined with real-time rainfall measurements, tide levels, and drainage capacity metrics. Each decision tree in the forest examines different features such as precipitation intensity, duration of rainfall, geographical elevation, proximity to water bodies, and historical flood frequency. The model classifies areas into distinct risk categories: Low Risk, Medium Risk, High Risk, and Critical Risk zones. By aggregating predictions from multiple trees (typically 100-500 trees), the Random Forest reduces overfitting and improves prediction accuracy compared to single decision trees. The ensemble approach ensures robustness against noisy data and provides confidence scores for each prediction.

2. KMeans Clustering:

KMeans Clustering is an unsupervised machine learning algorithm employed to group Mumbai's wards into distinct clusters based on their flood vulnerability characteristics. This technique enables the system to identify patterns and similarities among geographical areas without requiring pre-labeled training data.

The algorithm analyzes multiple geographical and infrastructural features including elevation levels, drainage density, historical flood frequency, population density, and proximity to coastlines or rivers. KMeans partitions wards into K clusters (typically 3-5 zones) where each cluster represents areas with similar flood vulnerability profiles. The algorithm iteratively assigns wards to clusters by minimizing the within-cluster variance, ensuring that wards within the same cluster share similar risk characteristics. Each cluster center (centroid) represents the average characteristics of that vulnerability zone. The resulting zones are labeled as: Critical Flood Zones, High Vulnerability Areas, Moderate Risk Regions, and Low Risk Districts.

3. Bayesian Network:

The Bayesian Network implements a probabilistic graphical model that captures complex dependencies and causal relationships between multiple environmental factors contributing to flood events. This advanced approach handles uncertainty inherently present in weather prediction and flood forecasting.

The network constructs a directed acyclic graph (DAG) where nodes represent variables such as rainfall intensity, tide levels, drainage system capacity, soil saturation, and flood occurrence. Directed edges between nodes represent probabilistic dependencies-for example, how high tide levels combined with heavy rainfall increase flooding probability. The model uses conditional probability tables (CPTs) to quantify the strength of relationships between variables. Given observed evidence (current rainfall, predicted tide levels, drainage status), the Bayesian Network performs probabilistic inference to calculate the likelihood of flooding in specific areas. The network updates its beliefs dynamically as new real-time data arrives, using Bayes' theorem to refine flood probability estimates.

4. A* Search Prediction:

The A* (A-star) Search Algorithm is a graph traversal and pathfinding algorithm that identifies the most optimal safe route between two locations while actively avoiding high-risk flood zones. This intelligent routing mechanism ensures user safety during flood emergencies.

Mumbai's road network is represented as a weighted graph where nodes are intersections/locations and edges are road segments. Each edge is assigned a dynamic risk weight based on real-time flood predictions-roads in high-risk zones receive higher costs, making them less desirable for routing. The algorithm uses a heuristic function $f(n) = g(n) + h(n)$, where $g(n)$ is the actual cost from the start node to current node n, $h(n)$ is the estimated cost from node n to the destination (using haversine distance), and the risk weight is incorporated into $g(n)$, increasing the cost of traversing flood-prone roads. A* explores the graph intelligently, always expanding the most promising node (lowest f-value) first. The algorithm guarantees finding the shortest safe path that balances distance minimization with flood risk avoidance. Routes are continuously recalculated if conditions change, ensuring navigation remains optimal as flood predictions update.

These technical specifications outline a comprehensive and forward-thinking approach, ensuring that the integration of advanced tools and methodologies allows SmartFlood Mumbai to deliver highly accurate predictions, real-time insights, and practical navigation assistance, setting a new standard in the domain of AI-driven risk mapping.

Chapter 7

Project Scheduling

In project management, a schedule is a listing of a project's milestones, activities, and deliverables. A schedule is commonly used in the project planning and project portfolio management parts of project management. The project schedule (Table 7.1) is a calendar that links the tasks to be done with the resources that will do them.

Sr.No.	Group Members	Duration	Task Performed
1.	Aditya Kate Tanmay Harmalkar Suman Manik	1st Week of July	Group formation and Topic finalization. Identifying the scope and objectives of the Mini Project. Discussing the project topic with the help of a paper prototype. Identifying the functionalities of the Mini Project.
2.	Suman Manik	2nd Week of July	Conducting literature review and training the machine learning models (Random Forest, K-Means, Bayesian Networks) based on historical flood datasets.
3.	Suman Manik	3rd Week of July	Designing the System Architecture and Graphical User Interface (GUI) using React and Leaflet.
4.	Aditya Kate Tanmay Harmalkar	1st Week of August	Implementing functional requirements like data integration, preprocessing, flood risk prediction, and safe route suggestion using A* algorithm.
5.	Aditya Kate Suman Manik	1st Week of September	Integrating the models with the frontend (React) and backend (FastAPI), and connecting APIs (e.g., OpenWeatherMap).
6.	Aditya Kate Tanmay Harmalkar	2nd Week of September	Testing system reliability, visualization dashboards, and overall project modules.

Table 7.1: Project Task Distribution

A Gantt chart is a type of bar chart that illustrates a project schedule. This chart lists the tasks to be performed on the vertical axis, and time intervals on the horizontal axis. Gantt chart (Fig 7.2) illustrates the start and finish dates of the terminal elements and summary elements of a project.

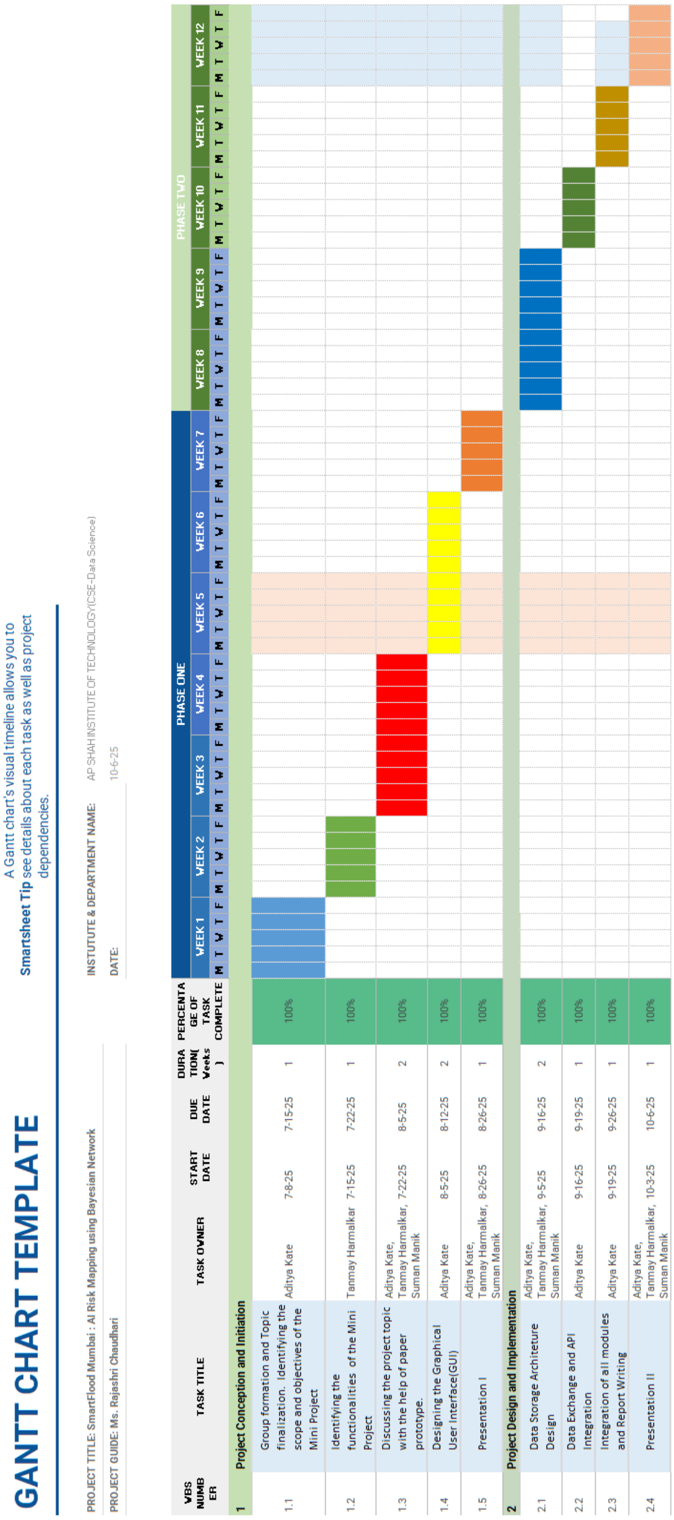

Table 7.2: Gantt Chart of SmartFlood Mumbai

Chapter 8

Results

The project results section provides a concise overview of the outcomes achieved through the implementation of the project. Highlighting key findings, deliverables, and the final implementation of the project lifecycle. This section serves to summarize the tangible outcomes and impacts of the project, providing stakeholders with valuable insights into its overall effectiveness and contribution to the intended objectives.

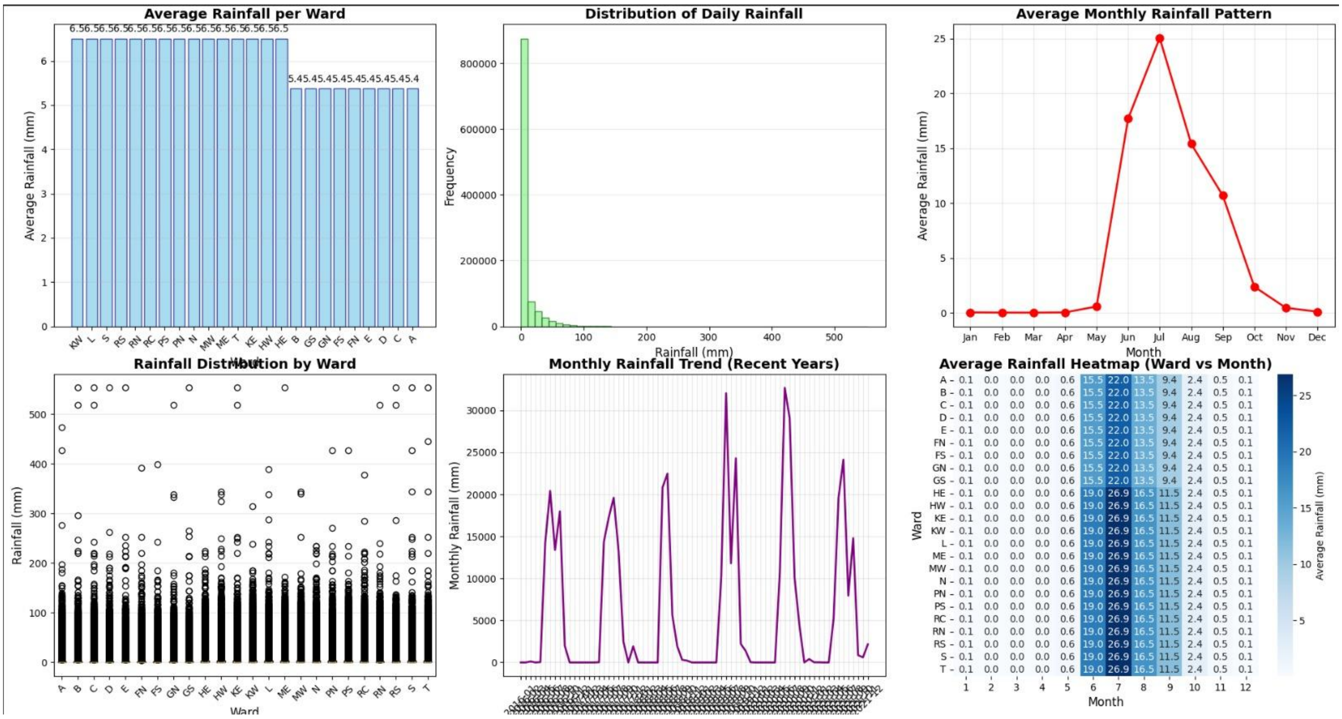

Fig 8.1: Analysis of Rainfall Patterns across various Wards in Mumbai

The figure 8.1 highlight Mumbai's rainfall patterns, showing mostly consistent average rainfall across wards with some variation. Daily rainfall is typically low but includes occasional heavy events. Monthly trends clearly peak during the monsoon season from June to September. Ward-wise data reveals local differences and occasional high rainfall spikes. Yearly trends show repeating monsoon peaks with some variability.

The heatmap confirms most rainfall occurs in monsoon months, varying slightly by ward. Overall, these findings support the need for detailed spatial and temporal rainfall analysis to improve flood risk predictions in SmartFlood Mumbai.

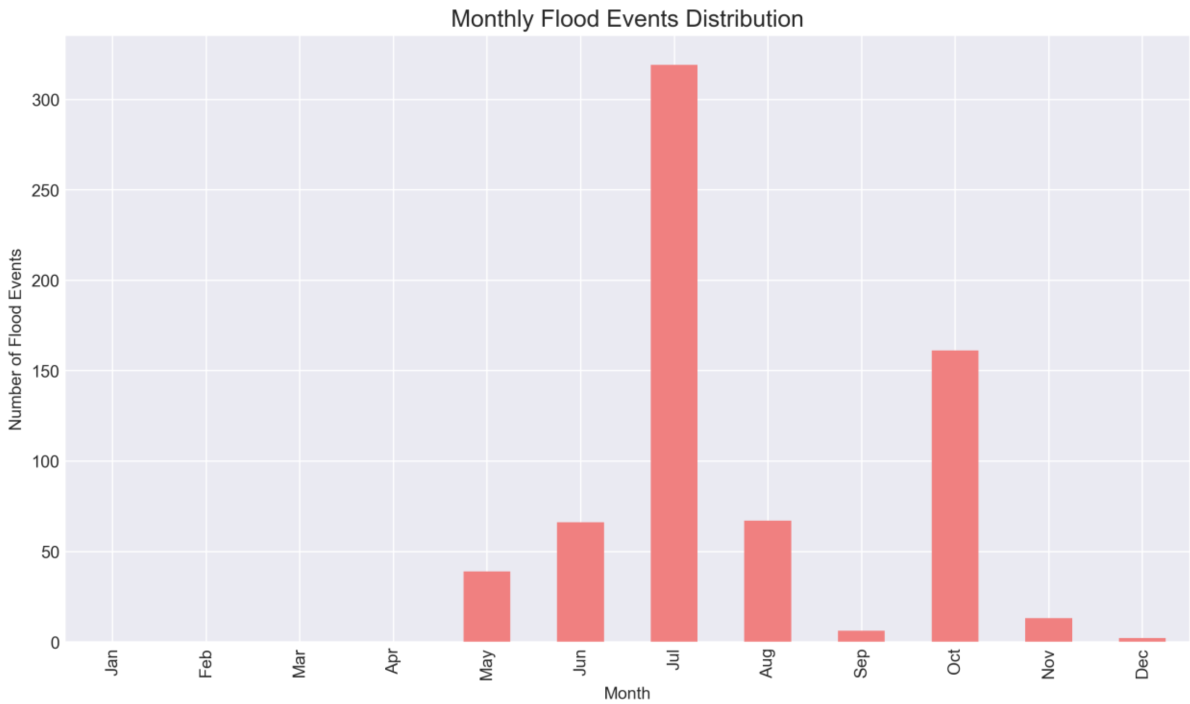

Fig 8.2: Monthly Distribution of flood events in Mumbai

The figure 8.2 illustrates the monthly distribution of flood events in Mumbai. The pattern shows a dramatic spike in flood occurrences during July, with over 320 events, making it the peak flood month. There is notable activity in May, June, August, and October, but July stands out as the most flood-prone period by a large margin. Minimal or almost no flood events are recorded in the other months, indicating that flooding is heavily concentrated in the monsoon season, with a sharp surge in July and smaller secondary peaks in late summer and early autumn. This highlights the critical need for heightened flood preparedness specifically around July, and provides clear temporal guidance for local authorities and citizens. The stark contrast between the monsoon months and the rest of the year, where flood events are virtually absent, allows citizens to plan their activities accordingly and take necessary precautionary measures during this high-risk period, while infrastructure resilience measures and evacuation protocols can be specifically calibrated to handle the concentrated burden during the July peak. This clear seasonality provides valuable insights for optimizing resource allocation, enabling local authorities to pre-position emergency services, conduct targeted public awareness campaigns, and implement preventive drainage maintenance well before the critical months arrive.

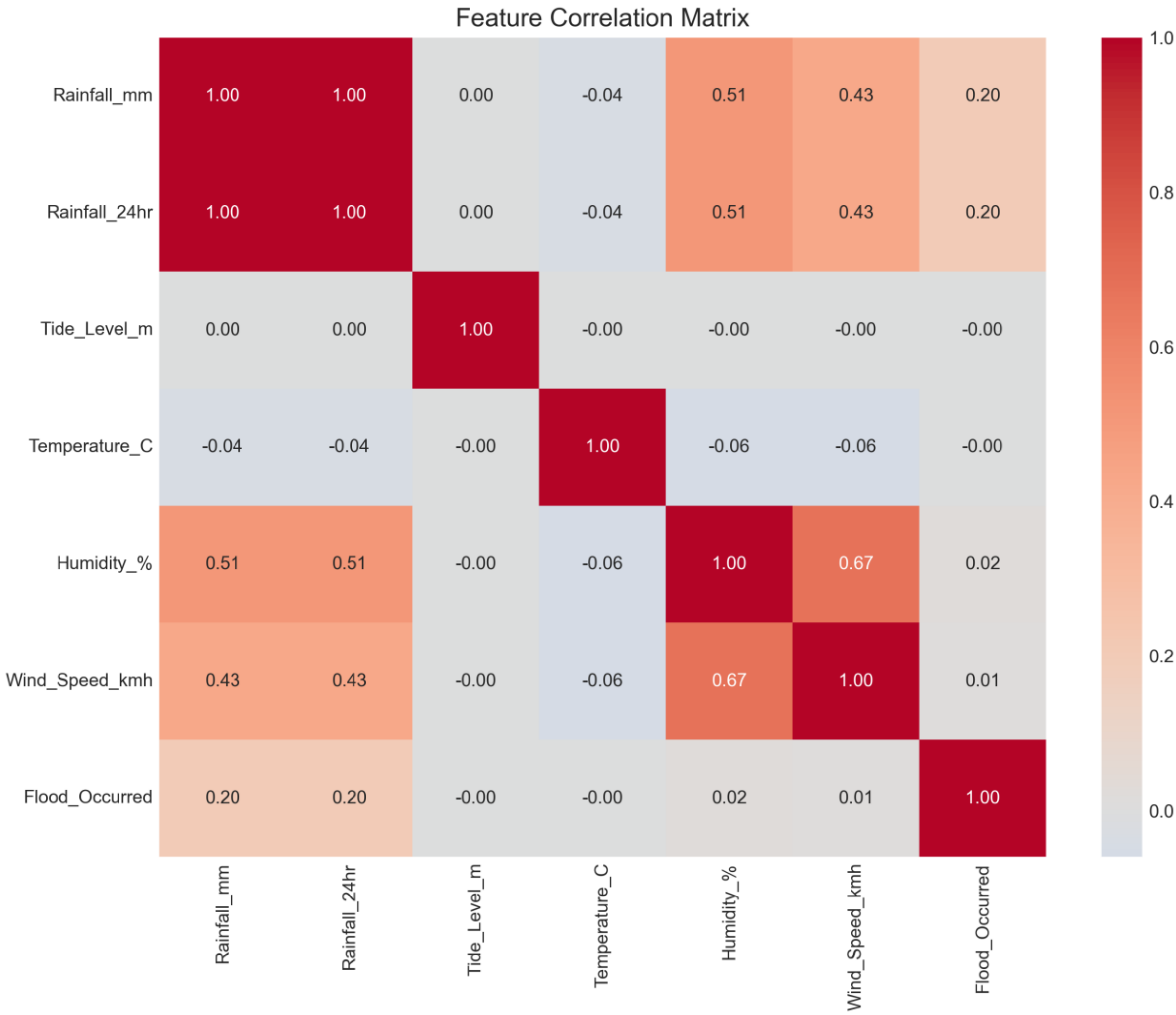

Fig 8.3. Feature Correlation Matrix of Meteorological Variables and Flood Occurrence in Mumbai

This figure 8.3 correlation matrix displays how different weather and environmental variables relate to each other and to flood events in Mumbai. Rainfall (both immediate and over the last 24 hours) shows perfect correlation with itself, and a moderately positive correlation with humidity and wind speed, indicating that wetter and windier conditions tend to coincide. Both rainfall measures have a weak but positive correlation with flood occurrence, meaning higher rainfall slightly increases the likelihood of floods, but is not the sole determining factor. Humidity and wind speed themselves are also moderately correlated, while tide level and temperature show little to no meaningful relationship with either rainfall or flood events. Overall, rainfall, humidity, and wind speed are the most relevant predictors of flooding in this analysis, whereas temperature and tide level appear to have minimal direct effect. This feature correlation matrix shows how different weather variables and flood occurrence are statistically related. Rainfall and 24-hour rainfall are perfectly correlated and have a moderate positive relationship with humidity and wind speed, suggesting wet and breezy conditions often go together. Both rainfall values have a weak positive correlation with flood occurred, meaning floods are more likely when it rains heavily, but other factors also play a role. Tide level and temperature show almost no correlation with floods or other weather features in this dataset. Overall, rainfall, humidity, and wind speed emerge as the most important predictors of flood events in Mumbai, while temperature and tide level are much less influential.

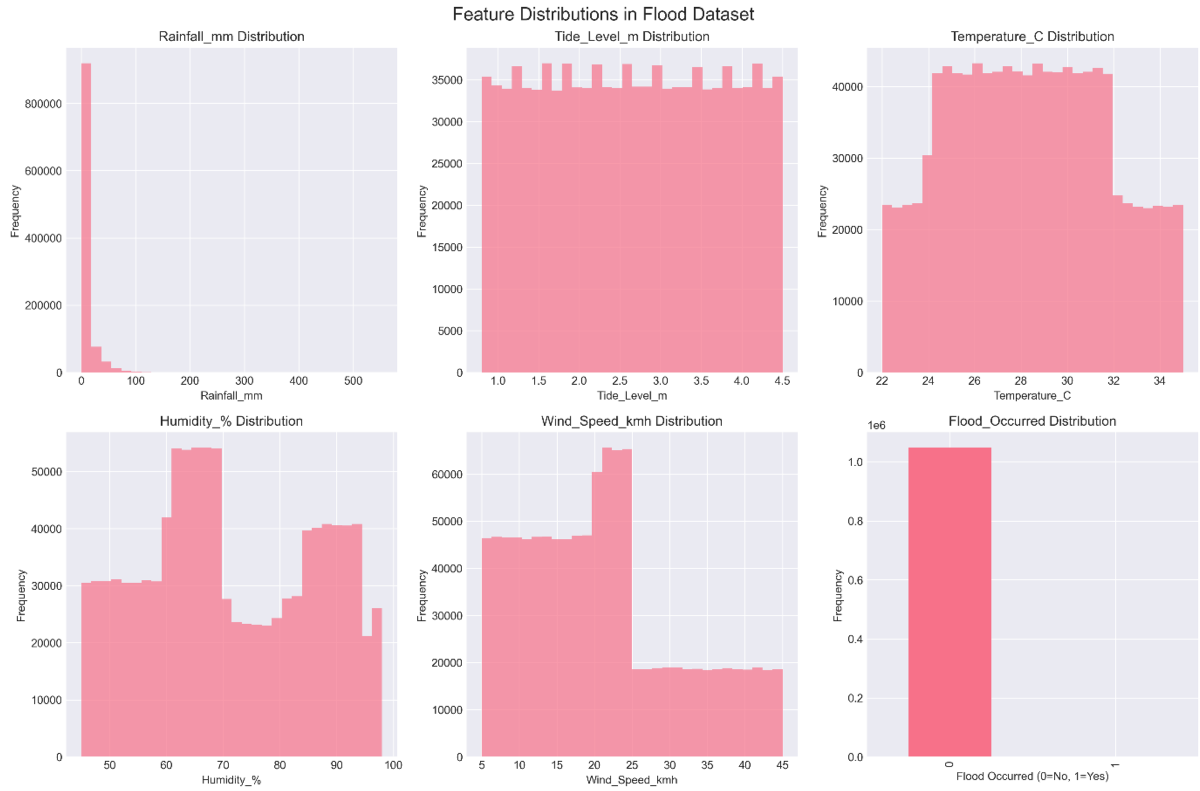

Fig 8.4. Feature Distribution of Flood in Mumbai Past

This figure presents the distributions of key meteorological and event features in the Mumbai flood dataset. The histogram for rainfall (Rainfall_mm) reveals that most days have low rainfall, but there are rare extreme values suggesting sporadic heavy rain events. The Tide_Level_m distribution appears even across its range, indicating no significant seasonal or directional skew in tide heights. Temperature_C is broadly distributed, with most observations centered around a moderate temperature range. The Humidity_% plot shows distinct peaks, suggesting periods of very high humidity are common, while Wind_Speed_kmh reveals a fairly uniform but slightly right-skewed distribution with many observations at lower speeds and fewer at higher speeds. Finally, the Flood_Occurred bar confirms that flood events are relatively scarce compared to non-flood occurrences, as most entries are "0" (no flood) with very few "1" values (flood). Together, these plots illustrate the variability within meteorological conditions in Mumbai and underscore the relative rarity of flood events compared to the volume of weather data collected.

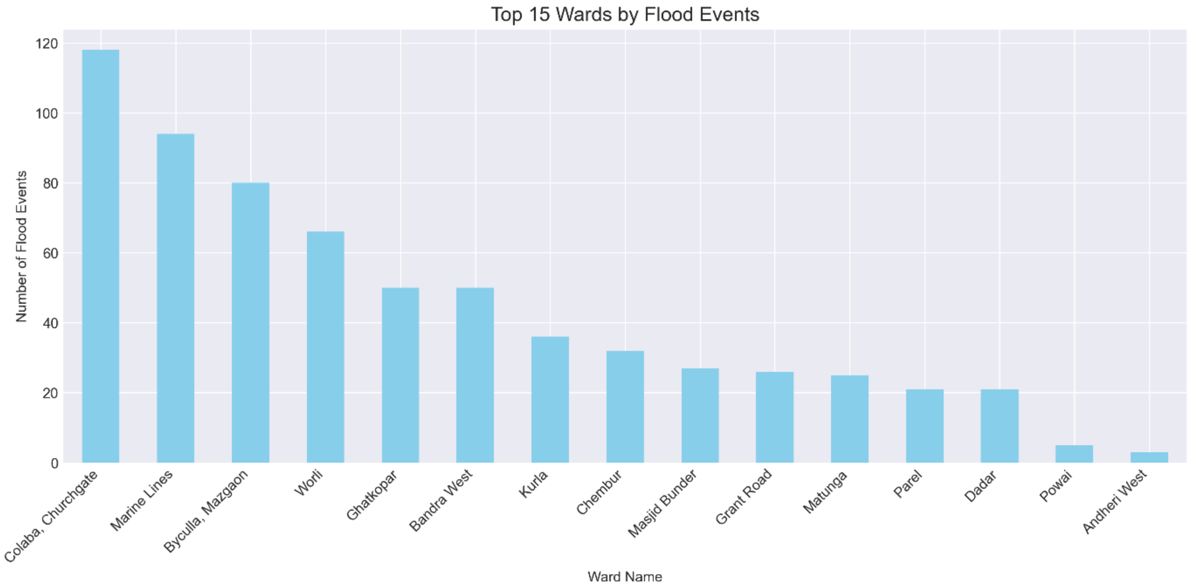

Fig 8.5. Feature Distribution in Flood Dataset

This figure 8.5 chart displays the top 15 wards in Mumbai with the highest number of flood events [2]. Colaba and Churchgate stand out with the most frequent floods, followed by Marine Lines, Byculla, and Mazgaon. Wards such as Worli, Ghatkopar, and Bandra West also show notable flood occurrences, while locations like Parel, Dadar, Powai, and Andheri West have comparatively fewer events. This ranking highlights areas within Mumbai that are most susceptible to flooding, underlining the importance of targeted flood management and prevention efforts in those high-risk wards.This chart shows which wards in Mumbai have experienced the most flood events. Colaba and Churchgate top the list, followed by Marine Lines, Byculla, and Mazgaon. Overall, the chart highlights that flooding is concentrated in a few wards, with places like Worli, Ghatkopar, and Bandra West also facing frequent incidents, while wards like Powai and Andheri West have much fewer reported floods. This information is crucial for focusing flood preparedness and mitigation strategies on the city’s most vulnerable wards, and allocation to reduce future flood impact.

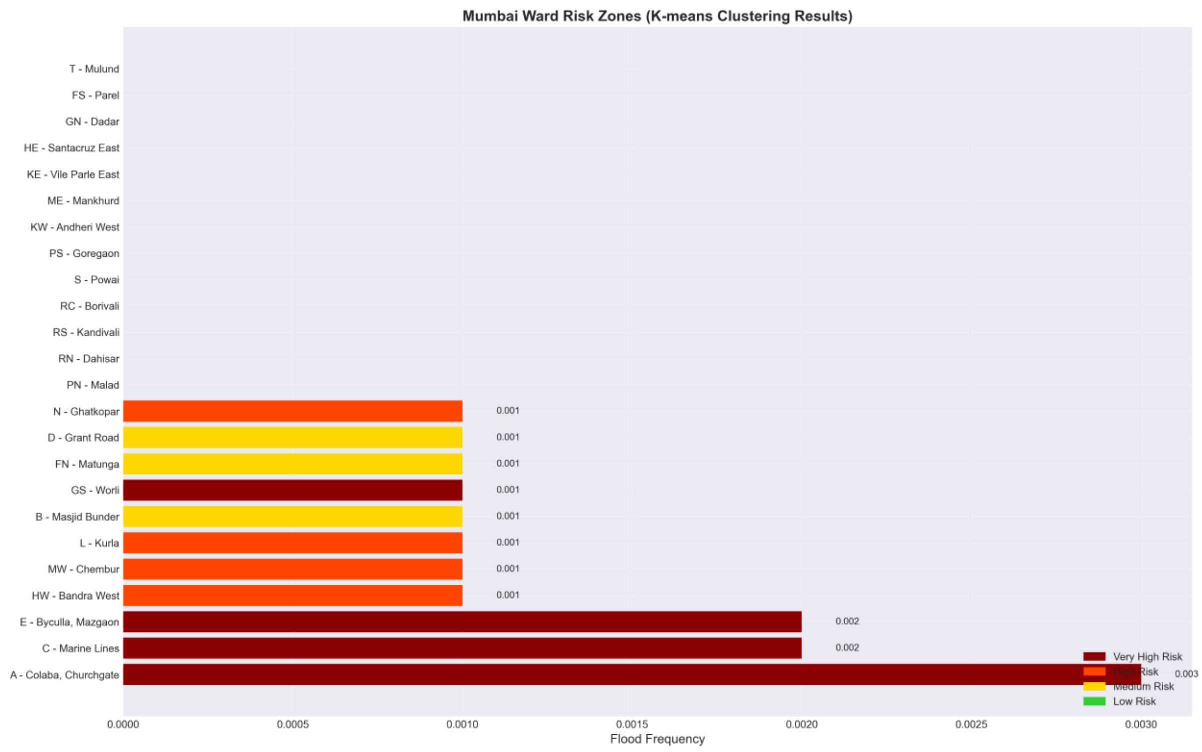

Fig 8.6. Mumbai Ward Flood Risk Zones Based on K-means Clustering of Flood Frequency

This figure 8.6. shows the flood risk classification of Mumbai wards based on K-means clustering of flood frequency data. Wards are grouped into risk zones represented by color: dark red for very high risk, red for high risk, orange for medium risk, and green for low risk. The x-axis indicates flood frequency, with higher values meaning more frequent floods.

Colaba and Churchgate stand out as the most vulnerable wards, classified as very high risk due to their highest flood frequencies. Marine Lines and Byculla, Mazgaon also fall within this very high-risk category. Several wards like Bandra West, Chembur, Kurla, and Masjid Bunder are categorized as high or medium risk, with moderate flood frequencies. Many other wards such as Mulund, Parel, Dadar, and Andheri West are on the low-risk side, showing very low flood frequencies or nearly zero flood events.

Overall, this visualization helps identify Mumbai's flood-prone wards and ranks them based on frequency and severity, enabling targeted flood management and mitigation in the highest risk zone.

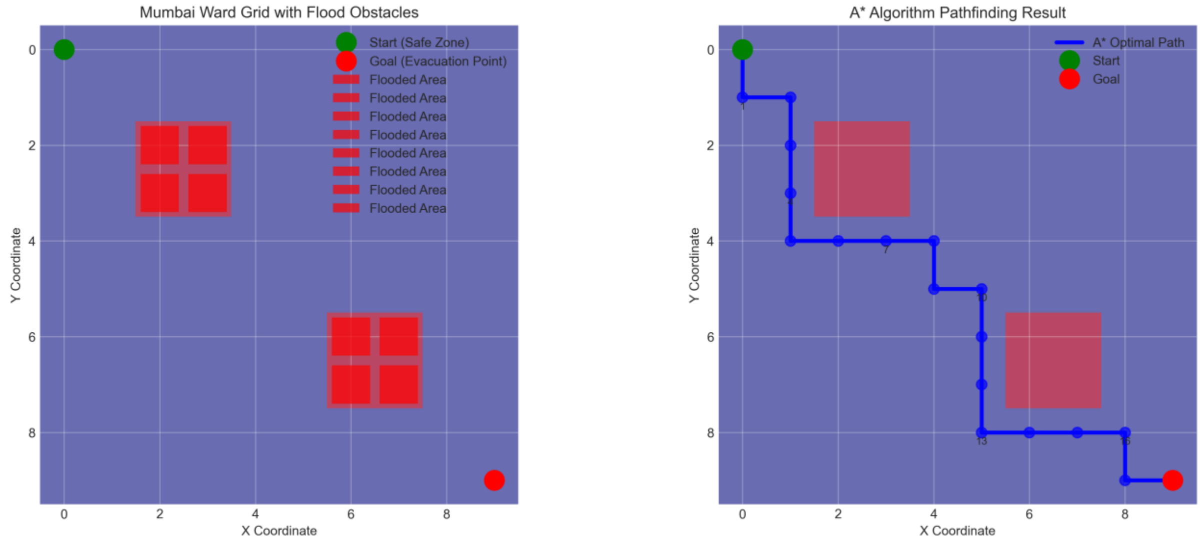

Fig 8.7. Flood-aware Pathfinding in Mumbai Ward Grid Using the A* Algorithm

This figure 8.7. demonstrates how the A* algorithm can be used for flood-aware route planning within a Mumbai ward grid featuring flood obstacles. The left plot, titled "Mumbai Ward Grid with Flood Obstacles," shows a simplified grid layout with designated start (safe zone, green dot) and goal (evacuation point, red dot) locations. Flooded areas are marked as red squares, representing zones that should be avoided during evacuation. The right plot, titled "A* Algorithm Pathfinding Result," overlays the grid with the computed optimal path (solid blue line and dots) that safely navigates from the start to the goal while circumventing flooded zones. This path demonstrates that the routing algorithm can efficiently guide evacuation even in the presence of significant obstacles, highlighting a practical application for real-time flood scenario evacuation planning.

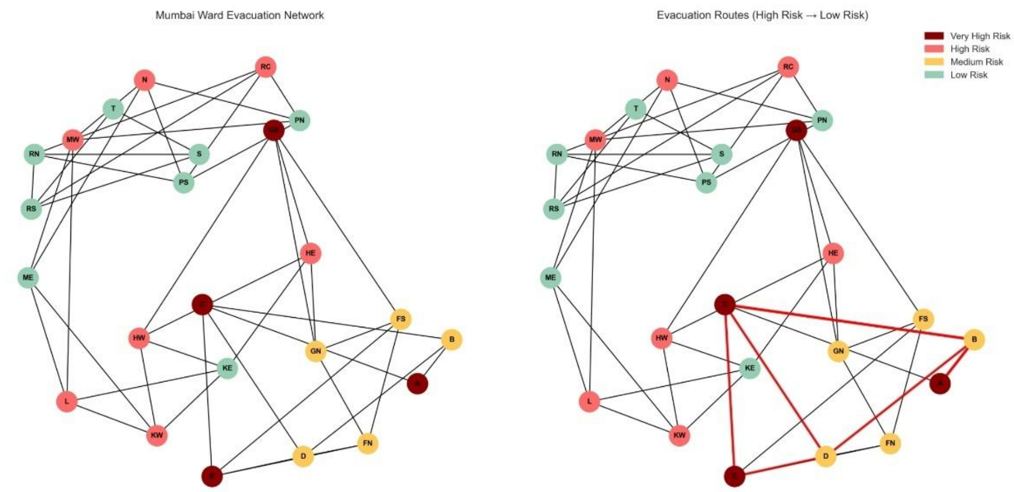

Fig 8.8. Mumbai Ward Flood Risk Evacuation Network and Optimal Safe Routes

This figure 8.8 visualizes the evacuation network and optimal evacuation routes between Mumbai wards based on their flood risk categories. The left graph, titled "Mumbai Ward Evacuation Network," depicts the connectivity between various wards as a network of nodes, each node representing a ward and colored according to its flood risk level: dark red (very high risk), red (high risk), yellow (medium risk), and green (low risk). The lines indicate possible evacuation pathways between wards.

The right graph, titled "Evacuation Routes (High Risk → Low Risk)," highlights the optimal evacuation paths from high-risk wards to safer, low-risk wards using bold red edges. This mapping directs emergency planners and residents toward the most secure evacuation routes during flood events, prioritizing movement from vulnerable areas to zones with lower flood risk. The color-coded nodes and highlighted paths make the flood risk hierarchy visually clear, demonstrating the value of risk-aware network analysis for disaster preparedness.

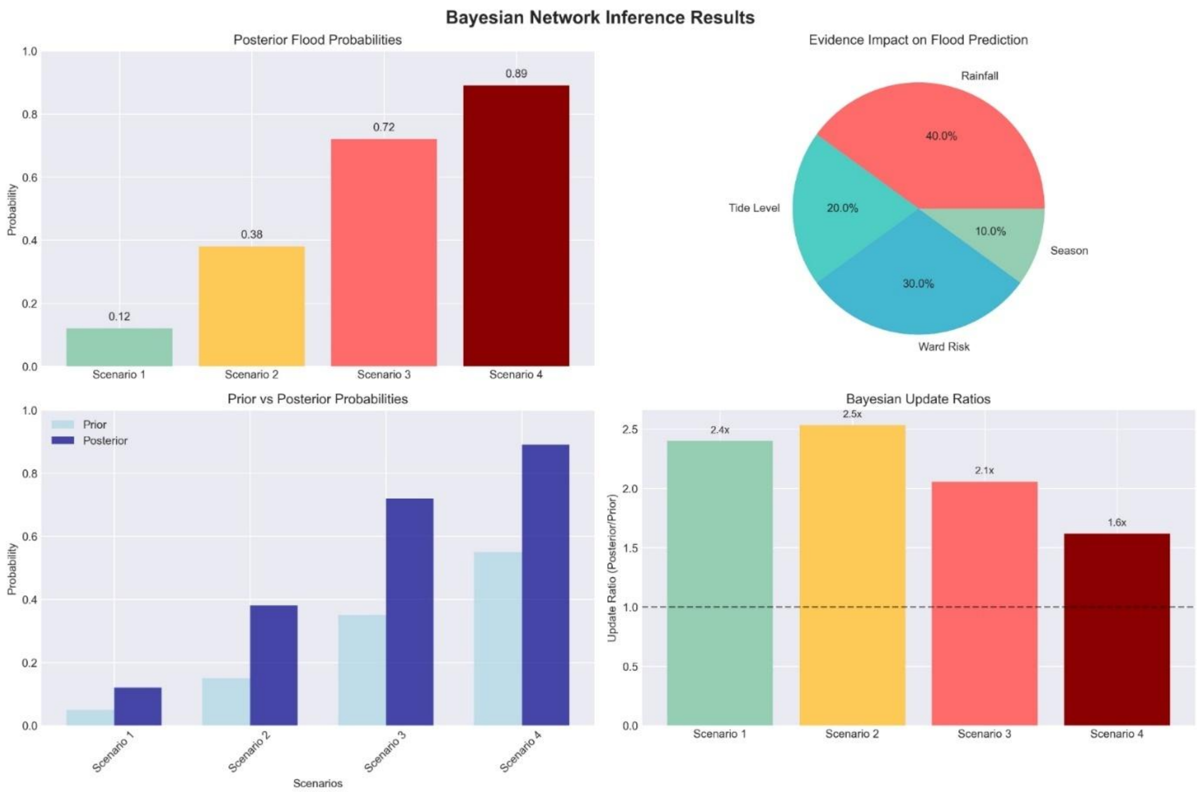

Fig 8.9. Bayesian Network Inference Results for Flood Prediction in Mumbai

This figure 8.9. summarizes the results of Bayesian Network inference applied to flood prediction in Mumbai. The top left bar chart, “Posterior Flood Probabilities,” illustrates how the likelihood of flood events increases across different scenarios, with Scenario 1 having the lowest probability and Scenario 4 the highest. The top right pie chart, “Evidence Impact on

Flood Prediction," breaks down the influence of input features, showing that rainfall contributes 40% of predictive power, ward risk 30%, tide level 20%, and season 10%. The bottom left chart, "Prior vs Posterior Probabilities," compares initial flood probabilities before evidence is considered (priors) against updated probabilities after incorporating evidence (posteriors), demonstrating how Bayesian reasoning refines predictions for each scenario. Lastly, the bottom right chart, "Bayesian Update Ratios," displays how much each scenario's probability increases after evidence is processed, quantifying the update effect as a multiplicative ratio.

These visualizations collectively highlight how the Bayesian network integrates multiple sources of evidence to significantly improve flood prediction accuracy, especially in scenarios with strong rainfall and high ward risk.

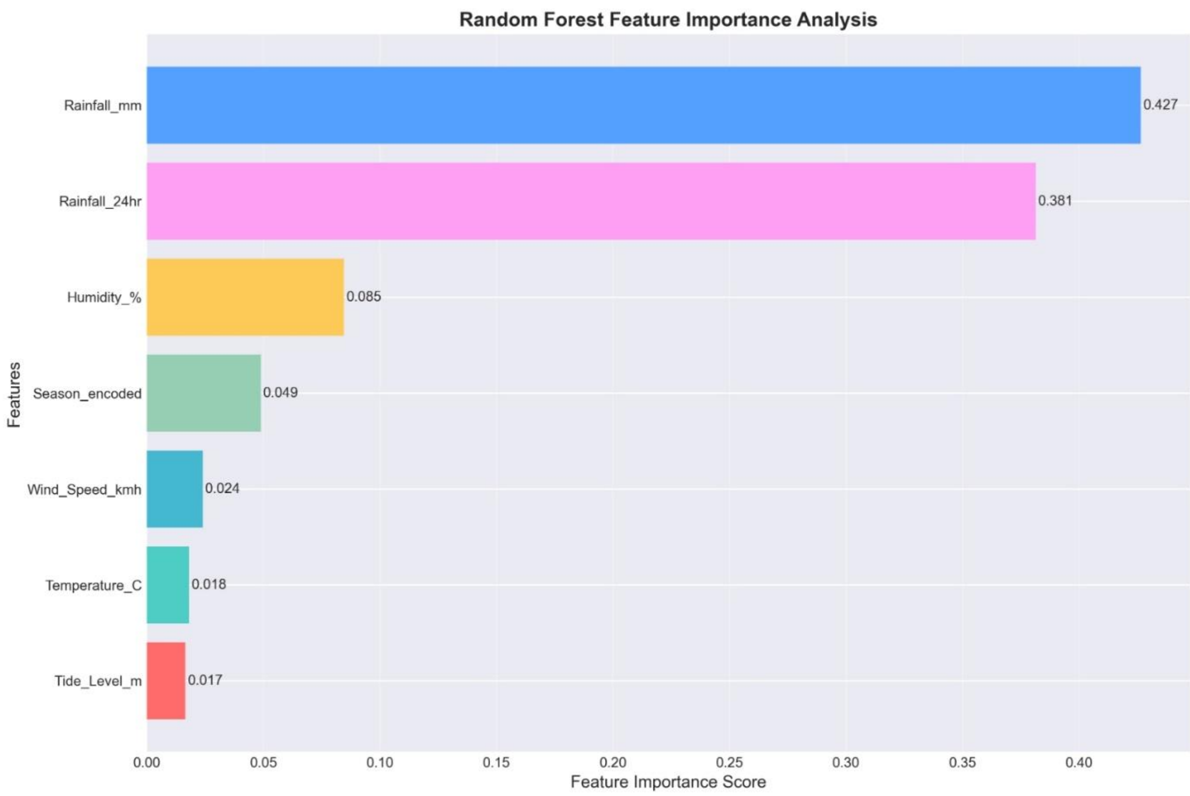

Fig 8.10. Random Forest Feature Importance Analysis for Flood Prediction in Mumbai

This figure 8.10 visualizes the importance of various meteorological and environmental features used by the Random Forest model to predict flood risk in Mumbai. The horizontal bar chart ranks features by their contribution to the model's decision-making process.

"Rainfall_mm" (current rainfall) is by far the most important factor, followed closely by "Rainfall_24hr" (rainfall in the past 24 hours). Humidity comes next, with a much smaller but still notable influence. Season, wind speed, temperature, and tide level each show relatively minor contributions to flood prediction, as reflected by their low importance scores This analysis highlights that immediate and recent rainfall dominate the model's flood risk calculations, emphasizing the central role of monsoonal dynamics in Mumbai's flood events, while other features play only a supporting role in predicting routes.

Chapter 9

Conclusion

The project was developed with the primary aim of delivering an efficient, reliable, and user-friendly system to address the critical issue of flood risk management in Mumbai. Through careful requirement analysis, system design, and implementation, the project successfully fulfills the objectives defined at the outset, providing a comprehensive solution that integrates predictive modeling, risk clustering, probabilistic reasoning, route optimization, and interactive visualization. The system not only streamlines flood-related decision-making processes but also significantly reduces manual errors and response delays, thereby enhancing the overall efficiency and effectiveness of disaster management operations.

In addition to its practical functionality, the project serves as a demonstration of how technology and artificial intelligence can be applied to solve real-world urban challenges in a structured and systematic manner. The integration of multiple AI algorithms-Random Forest, K-Means clustering, Bayesian Networks, and the A* search algorithm-highlights the potential of combining diverse computational techniques into a single, cohesive platform that produces actionable insights for multiple stakeholders.

The development of this project provided a valuable opportunity to bridge theoretical knowledge with practical implementation, offering hands-on experience in software development, machine learning, data processing, and geospatial visualization. Each stage of the project, from requirement gathering and system modeling to testing and deployment, was carried out methodically to ensure system stability, accuracy, and usability. The iterative design and testing process also emphasized the importance of user-centric development, ensuring that both technical and non-technical users can effectively interact with the system.

Overall, the project achieves its intended purpose of enhancing flood preparedness and decision-making at the ward level. Beyond its immediate application, it stands as a scalable and adaptable framework that can be further developed to accommodate additional features, expanded datasets, or deployment in other flood-prone cities. The project thus not only delivers a functional tool for Mumbai but also contributes to broader learning and innovation in the fields of AI, urban disaster management, and intelligent systems design.

Chapter 10

Future Scope

Expanding SmartFlood Mumbai to include cloud integration, advanced analytics, mobile extensions, and other features can significantly enhance its value proposition. Here are some potential future scope ideas.

1. Cloud Integration:

Implementing cloud services would enable remote access, improve scalability, and support real-time data processing for multiple users, allowing the system to handle larger datasets and facilitate distributed collaboration among stakeholders.

2. Advanced Analytics and Notifications:

Incorporating predictive trend analysis, risk scoring over time, and automated alerts via email, SMS, or push notifications could provide proactive guidance, with detailed reporting features summarizing flood trends and high-risk zones for better decision-making.

3. Security and Access Control:

Adding robust user authentication, role-based access control, and encrypted data storage would protect sensitive information like ward-level risk data and emergency routes from unauthorized access or misuse.

4. Mobile Platform Extension:

Developing a mobile application could deliver real-time flood updates and safe route suggestions on smartphones, including location-aware alerts based on users' current positions for enhanced accessibility.

5. IoT and Advanced AI Integration:

Integrating IoT sensors for real-time water-level monitoring and employing sophisticated AI techniques like deep learning could improve predictive modeling, while enabling multi-city scalability to transform the platform into a comprehensive urban flood management solution.

References

[1] Tripathy, S.S., Chaudhuri, S., Murtugudde, R., Mhatre, V., Parmar, D., Pinto, M., Zope, P.E., Dixit, V., Karmakar, S., & Ghosh, S. (2024). "Analysis of Mumbai floods in recent years with crowdsourced data." Journal of Hydrology and Hydromechanics, 72(1), 95-108. https://www.sciencedirect.com/science/article/abs/pii/S2212095524000117

[2] Yash, P., Solanki, Vijendra, K., Kul Vaibhav, S., Arpan, D., & Deepak, T. (2024). "Flood hazard analysis in Mumbai using geospatial and multi-criteria decision-making techniques." Journal of Water and Climate Change, 15(5), 2484-2502. https://iwaponline.com/jwcc/article/15/5/2484/101820/Flood-hazard-analysis-in-Mumbai-using-geospatial

[3] Pathak, S., Liu, M., Jato-Espino, D., & Zevenbergen, C. (2020). "Social, economic and environmental assessment of urban sub-catchment flood risks using a multi-criteria approach: A case study in Mumbai City, India." *Journal of Hydrology*, 591, 125216. https://www.sciencedirect.com/science/article/abs/pii/S0022169420306764

[4] Prajapat, P., Joshi, M., Yadav, S., Parmar, H., Ahmed, S., et al. (2025). "Analysis of urban flood vulnerability at the sub-city scale: empirical evidence from Mumbai, India." *Journal of Hydrology*"
https://www.sciencedirect.com/science/article/abs/pii/S2212095525002524

[5] Ali, H., Modi, P., Mishra, V., et al. (2024). "Flood risk assessment for Indian sub-continental river basins." Hydrology and Earth System Sciences, 28, 1107-1132. https://hess.copernicus.org/articles/28/1107/2024/

[6] OpenCity, "Mumbai Wards Map," Mumbai Open Data Portal. [Online]. Available: https://data.opencity.in/dataset/mumbai-wards-map. [Accessed: Dec. 26, 2024].

[7] OpenCity, "Mumbai Rainfall Data," Mumbai Open Data Portal. [Online]. Available: https://data.opencity.in/dataset/mumbai-rainfall-data. [Accessed: Dec. 26, 2024]