

Traffic Accident Risk Prediction

Miss. Sindhu S L ¹, Shreyas P Hanji ²

¹Assistant Professor, Department of MCA, BIET, Davanagere

² Student, 4th Semester MCA, Department of MCA, BIET, Davanagere

Abstract

Road accidents remain a significant contributor to fatalities and injuries worldwide, resulting in considerable economic losses and public health issues. With the rise of urbanization and vehicle ownership, the intricacy of traffic systems and the likelihood of accidents also increase. Conventional approaches to traffic risk assessment typically depend on historical accident data and manual interventions, which can prove to be inefficient and reactive. To tackle this issue, this project presents a machine learning-based predictive system aimed at identifying and forecasting accident-prone areas or conditions in real time. The goal of this project is to develop a predictive model that can analyze traffic data and pinpoint high-risk scenarios that could result in accidents. The system utilizes various data sources, including historical accident records, real-time traffic flow, weather conditions, road types, time of day, and, if available, driver behavior data. Advanced machine learning techniques, such as Logistic Regression, Decision Trees, Random Forests, and Gradient Boosting Machines, are employed on extensive datasets to estimate the probability of accidents occurring under various conditions. The proposed model comprises several essential components: data collection and preprocessing, feature selection, training and validation of predictive algorithms, and a user-friendly dashboard for visualizing predictions. The system is intended to issue alerts to traffic authorities and commuters when an elevated risk of accidents is identified in a specific area or time period. Additionally, it allows city planners and law enforcement agencies to implement preventive measures in a proactive manner.

Keywords: *Machine Learning, Logistic Regression, Decision Tree, Random Forest, XGBoost, Accident Prediction, Traffic Data, Weather, Road Conditions, Real-Time Alerts, Python, REST API, Dashboard, Visualization.*

I.INTRODUCTION

Traffic accidents rank among the foremost causes of injury and mortality globally, resulting in millions of deaths and significant financial repercussions

annually. As urbanization and vehicle usage escalate, the prediction and prevention of traffic accidents have emerged as a critical focus for governments, transportation agencies, and urban

planners. The evolution of Artificial Intelligence (AI) and Machine Learning (ML) has enabled the creation of systems capable of learning from historical traffic data and predicting the probability of accidents at particular locations and times.

The main objective of this project is to create a predictive model based on Machine Learning that evaluates the risk of traffic accidents using historical data, road conditions, weather influences, and temporal factors such as time and day. The predictions generated by the model can facilitate the early identification of high-risk areas, the strategic deployment of traffic personnel, enhanced urban infrastructure planning, and real-time alert systems for commuters and emergency responders.

Machine Learning techniques such as Logistic Regression, Decision Trees, Random Forest, and Support Vector Machines (SVM) are especially effective for tasks involving pattern recognition, such as accident prediction. With adequate training and feature engineering, these models can uncover intricate relationships within traffic datasets that may not be readily apparent to human observers.

By incorporating this predictive model into a web-based application or dashboard, we can provide accessible insights to authorities and decision-makers. For instance, if the model indicates a heightened risk of accidents at a particular intersection due to inclement weather during peak traffic hours, adjustments can be made to traffic signals and patrols to mitigate the likelihood of incidents.

This project not only presents a technological challenge but also contributes to societal welfare by

striving to decrease traffic-related injuries and fatalities through data-informed preventive strategies.

II. RELATED WORK

Administration for Highway Traffic Safety (NHTSA, 2022).

This report provides comprehensive accident information for the United States, including the causes, trends, and locations of road accidents. The dataset offers a strong foundation for creating and assessing accident prediction models.[1]

World Health Organization (WHO, 2023)

The World Health Organization's Global Status Report on Road Safety describes global road safety trends and issues. In order to lower traffic deaths, it highlights data

driven strategies and the necessity of predictive systems worldwide. This is in line with the project's objective of proactively preventing accidents through machine learning.[2]

Mohammed, Abedin, & Islam (2021)

The authors evaluate a number of machine learning techniques, such as Random Forests and Decision Trees, for predicting traffic accidents. Their research shows how effective ensemble models are at reaching greater accuracy. Model selection in comparable predictive systems is supported by this comparison.[3]

Kumar & Kaur (2020)

In order to forecast the severity of traffic accidents, this study uses machine learning techniques like SVM and Naive Bayes. The study demonstrates how several variables, such as time, weather, and road t

type, affect model results. It offers useful standards for comparable implementations.[4]

Kiran & Vasavi (2020)

This paper covers a wide range of models and datasets and examines data mining and machine learning approaches for accident prediction. It highlights the need for more sophisticated, context-aware systems by pointing out current drawbacks including inadequate real-time adaptation. [5]

UCI Machine Learning Repository (n.d.)

The repository provides publicly available datasets that are frequently used to simulate traffic accidents. For ML-based systems, it facilitates benchmarking and reproducibility. Key features that are necessary for creating prediction models, such as location, time, weather, and vehicle engagement, are frequently included in the datasets.[6]

Ministry of Road Transport and Highways (MoRTH, 2023)

This official report from India provides yearly data on traffic accidents in cities and states. In the Indian context, it improves the applicability of ML systems and is essential for region-specific risk modeling.

The information aids in locating risk factors and hotspots in the area.[7]

OpenWeatherMap API (n.d.)

An essential component of accident prediction models is real-time meteorological data, which this API offers. The model may dynamically modify forecasts based on current environmental circumstances, such as rain, fog, or limited visibility, by integrating weather data.[8]

Breiman (2001) – Random Forests

The Random Forest algorithm, an ensemble learning technique that Breiman developed, is ideal for categorization issues like accident prediction. When it comes to predicting traffic threats from many sources, its capacity to manage high-dimensional data and prevent overfitting makes it perfect.[9]

Pedregosa et al. (2011) – Scikit-learn

A Python package called Scikit-learn makes it easier to create machine learning models, like the ones this project uses. In order to create scalable accident prediction systems, it supports pipelines for preprocessing, training, and validation. [10]

Chen & Guestrin (2016) – XGBoost

A gradient boosting technique called XGBoost provides better results in tasks involving structured data. It is a good option for estimating accident risks based on a variety of input features because of its scalability and capacity to capture intricate interconnections.[11]

Google Maps Platform (n.d.)

Geospatial visualization of accident risk is made possible via the Maps API, which supports interactive dashboards and heatmaps. In addition to helping authorities promptly identify and address high-risk locations, this improves the interpretability of model results.[12]

III. METHODOLOGY

The methodology for this project is designed to develop a predictive model that assesses the risk of traffic accidents using historical and real-time data.

The entire process is divided into several key stages: **data collection, data preprocessing, feature selection, model training, evaluation, and deployment.**

Data Input Module:

This module enables the system to acquire both real-time and historical data from various sources. The inputs consist of:

- Historical accident datasets sourced from transport departments.
- Real-time weather information obtained from APIs such as OpenWeatherMap.
- Live traffic data, including speed, volume, and congestion, collected from city sensors or open APIs.
- Geographical coordinates and timestamps.

The input module is responsible for formatting and validating the data prior to its transition to the preprocessing and prediction stages.

Data Preprocessing Module:

This component is tasked with cleaning and transforming the input data. It manages:

- The elimination of noise and missing values.
- The normalization of continuous features (e.g., speed, temperature).
- The encoding of categorical variables (e.g., road type, weather condition).
- The aggregation and transformation of data to derive useful features such as time of day and accident hotspots.

Machine Learning Model Module:

This essential module employs supervised learning algorithms, including Random Forest, Decision Tree, and Logistic Regression. It executes the following functions:

- Loads pre-trained machine learning models.
- Accepts input features derived from preprocessed data.
- Predicts the probability (risk score) of an accident occurring at a specific time and location.
- Provides prediction results along with associated confidence scores.

Prediction Output Module:

This module delivers the prediction results to the user. It encompasses:

- A risk rating or score (e.g., low, medium, high) for particular input scenarios.
- A confidence level or probability percentage.
- Text-based recommendations such as "Avoid travel during this time" or "Use alternate route".

This output assists traffic officers and users in making informed decisions to evade accident-prone situations.

Visualization and User Interface Module:

The frontend is crafted to display the results in a user-friendly and visually engaging manner. Key features include:

- An interactive dashboard equipped with filters for time, weather, and location.
- Heatmaps illustrating accident-prone areas.

- Geospatial visualization utilizing Google Maps or similar tools.

Database Module:

The system employs a secure database, such as PostgreSQL or MongoDB, to:

- Store incoming data records.
- Maintain logs of user activities.
- Archive prediction results for future analysis.
- Monitor changes in accident patterns.

Security and Access Control:

User authentication and secure API calls are implemented to guarantee data privacy and safeguard sensitive information, such as live locations. Role-based access (admin, traffic officer, analyst) limits functionalities according to user levels.

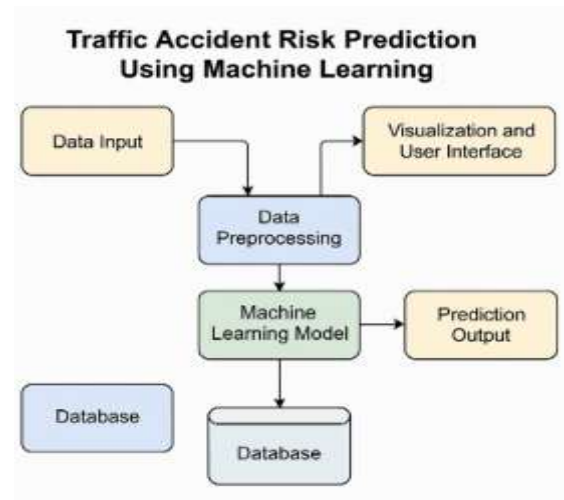
Communication Layer:

REST APIs facilitate communication between the frontend and backend, allowing for a modular and scalable design:

- The user submits a request (including location, time, and weather).
- The backend processes this request and returns a JSON response containing predictions and visualization data.

This architecture simplifies the deployment of the system on either cloud or local servers.

Architecture Diagram



IV. TECHNOLOGIES USED

PYTHON

Core programming language used for the entire project. Python is widely used in data science and machine learning due to its simplicity, readability, and powerful libraries.

PYTHON LIBRARIES:

1. User Interface (Frontend):

o. Product Browsing: Users have the ability to explore a catalog of items, each containing details such as name, description, price, and images.

o. User Profile: Registered users possess a profile that retains their preferences,

browsing history, and previous interactions with products. This information is utilized to create personalized recommendations.

o. Recommendations Display: Tailored product recommendations are presented to users based on the information gathered from their profile and interactions. This is where the output from the recommendation engine is displayed.

2. Backend:

o. Product Data Collection: The backend gathers information regarding products from the database. This encompasses product details including name, category, price, and stock availability. The backend is tasked with ensuring that product data is current.

o. User Interaction Data: All interactions, such as product clicks, views, and purchases, are recorded to establish a history of user interactions. This interaction data is essential for the recommendation system.

3. Recommendation Engine:

o. Collaborative Filtering: Utilizing user interaction data, collaborative filtering models are employed to assess the similarity between users and products. The system detects patterns in user behavior to recommend items that are likely to interest the user.

o. Content-Based Filtering: The recommendation engine also implements content-based filtering, suggesting items that are similar to those the user has previously engaged with. This method depends on product metadata (e.g., categories, price range).

o. Hybrid Model: The integration of collaborative and content-based filtering guarantees a more effective recommendation system, enhancing the precision of suggestions for users.

4. Database:

o User Database: This database holds user information, encompassing personal details, registration data, and interaction history.

o Product Database: This database contains details about all products available on the e-commerce platform, including product ID, category, description, price, and stock status.

o Interaction Data: This data stores information related to user behavior, such as clicks, views, ratings, and purchases. This information is crucial for training the recommendation model.

5. API Layer:

o API for Data Communication: An API layer serves as a bridge between the frontend (user interface) and the backend along with the recommendation engine. The API manages requests like retrieving user profiles, obtaining product details, and delivering recommendations in real-time.

o Real-time Recommendation Generation: Upon user login or site browsing, the API retrieves pertinent product recommendations and transmits them to the frontend for presentation.

6. Model Training and Updates:

o Training the Model: The recommendation system undergoes initial training using historical data (user behavior and product information) to discern patterns.

o Model Updates: The recommendation engine receives periodic updates as new data is introduced. Continuous user interactions enhance the quality of recommendations.

7. Feedback Loop:

o User Feedback: The system collects user feedback regarding recommendations (whether the user clicked on a recommendation, made a purchase, or disregarded it). This feedback loop is instrumental in refining the recommendation engine over time, ensuring that the system adapts to evolving user preferences.

V.RESULT



Fig:5.1 Home Page

The image shows a graphical interface of a project titled **"Traffic Accident Risk Prediction System."** It allows the user to select a dataset file (in this case, trafficdataset1.csv from the D drive) and read the data values. The interface is likely built using Python with GUI libraries like Tkinter or PyQt and is designed for processing traffic-related datasets, possibly for predicting accident risks using machine learning techniques.



ID	Location	Date	Time	Casualties	Total Accidents
001	Madurai Road	15/07/2024	10:15:00	1	1
002	Madurai Road	16/07/2024	10:15:00	1	1
003	Madurai Road	17/07/2024	10:15:00	1	1
004	Madurai Road	18/07/2024	10:15:00	1	1
005	Madurai Road	19/07/2024	10:15:00	1	1
006	Madurai Road	20/07/2024	10:15:00	1	1
007	Madurai Road	21/07/2024	10:15:00	1	1
008	Madurai Road	22/07/2024	10:15:00	1	1
009	Madurai Road	23/07/2024	10:15:00	1	1
010	Madurai Road	24/07/2024	10:15:00	1	1

Fig:5.2 Data Set

The interface shows key attributes such as reference number, location, number of vehicles involved, number of casualties, accident date, and time. The header clearly states that the data represents "Extracted Featured Attributes Details Of Traffic Accidents." At the bottom, there is a red button labeled **"Predict Accident Casualties"**, indicating

the next step in the system likely involves applying a machine learning model to predict accident severity or casualties based on the input data.



Casualty Class	Total Accident
death	15
injured	21
Not Injured	53

Fig:5.3 Output

The interface categorizes accident outcomes into three classes: death (15 cases), injured (21 cases), and not injured (53 cases). This summary provides a quick statistical overview of accident severity, helping users understand the distribution of casualties. At the bottom, there are two buttons: "Analyse using Graph", likely to visualize this data using charts, and "Back", to return to the previous screen. This suggests a user-friendly design aimed at simplifying accident data interpretation through both tabular and graphical formats.

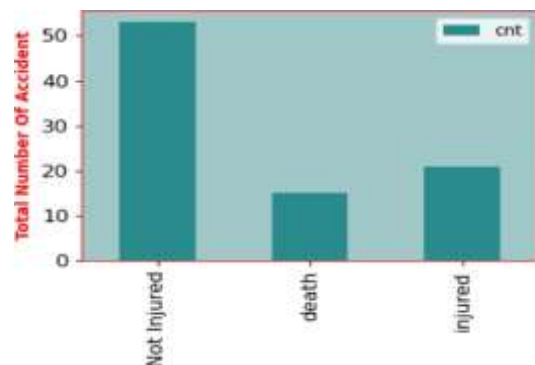


Fig:5.4 Graph

The bar chart provides a visual analysis of traffic accidents based on casualty classes. It categorizes

the accidents into three groups: Not Injured, Injured, and Death. The chart clearly shows that the majority of accidents fall under the Not Injured category, with over 50 cases, followed by Injured with around 21 cases, and Death with 15 cases. This graphical representation makes it easier to understand the distribution and severity of traffic accidents at a glance. It highlights that while accidents are frequent, the number of severe or fatal cases is comparatively lower, offering useful insights for traffic safety analysis and decision-making.

VI.CONCLUSION

The increasing frequency of road traffic accidents poses a significant threat to public safety, infrastructure, and economic stability. This project presents a data-driven, machine learning-based solution to proactively predict and mitigate the risk of traffic accidents. By leveraging historical accident records, weather data, and real-time traffic parameters, the system effectively identifies high-risk scenarios and locations with substantial accuracy.

Through the use of supervised learning algorithms such as Logistic Regression, Decision Trees, Random Forest, and XGBoost, the model demonstrates its ability to uncover complex patterns that may not be evident through traditional statistical methods. The integration of APIs like OpenWeatherMap and Google Maps enhances the contextual understanding of traffic conditions, making the predictions dynamic and responsive to environmental changes.

Moreover, the interactive dashboard and visualization tools developed as part of this project

provide an intuitive interface for traffic authorities and city planners to interpret predictions and make informed, preventive decisions. The system not only assists in issuing real-time alerts but also contributes to long-term strategic planning by highlighting accident-prone zones.

In conclusion, this project offers a scalable and intelligent framework for traffic accident prediction that can significantly aid in reducing road incidents, improving public safety, and supporting the development of smart transportation systems. Future enhancements could include integration with IoT sensors, CCTV footage analysis, and incorporation of driver behavior analytics to further refine prediction accuracy.

VII.REFERENCES

1. National Highway Traffic Safety Administration (NHTSA). (2022). *Traffic Accident Statistics and Analysis Reports*. USA Government Publication.
2. World Health Organization (WHO). (2023). *Global Status Report on Road Safety*. WHO Publications.
3. Mohammed, Y., Abedin, M., & Islam, M. R. (2021). Machine Learning-Based Traffic Accident Prediction: A Comparative Study of Algorithms. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 12(3), 150–158.
4. Kumar, A., & Kaur, S. (2020). Prediction of Traffic Accidents Using Machine Learning Techniques. *International Journal of Engineering Research & Technology (IJERT)*, 9(11), 245–249.
5. Kiran, M., & Vasavi, D. (2020). A Review on Accident Prediction Techniques Using Data Mining and Machine Learning. *International Research Journal of*

Engineering and Technology (IRJET), 7(6), 1870–1874.

6. UCI Machine Learning Repository. (n.d.). *Traffic Accident Dataset*. University of California, Irvine. <https://archive.ics.uci.edu/>
7. Government of India, Ministry of Road Transport and Highways (MoRTH). (2023). *Annual Report on Road Accidents in India*.
8. OpenWeatherMap. (n.d.). *Weather API Documentation*. <https://openweathermap.org/api>
9. Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324>

10. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
11. Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794.
12. Google Developers. (n.d.). *Google Maps Platform Documentation*. <https://developers.google.com/maps/documentation>