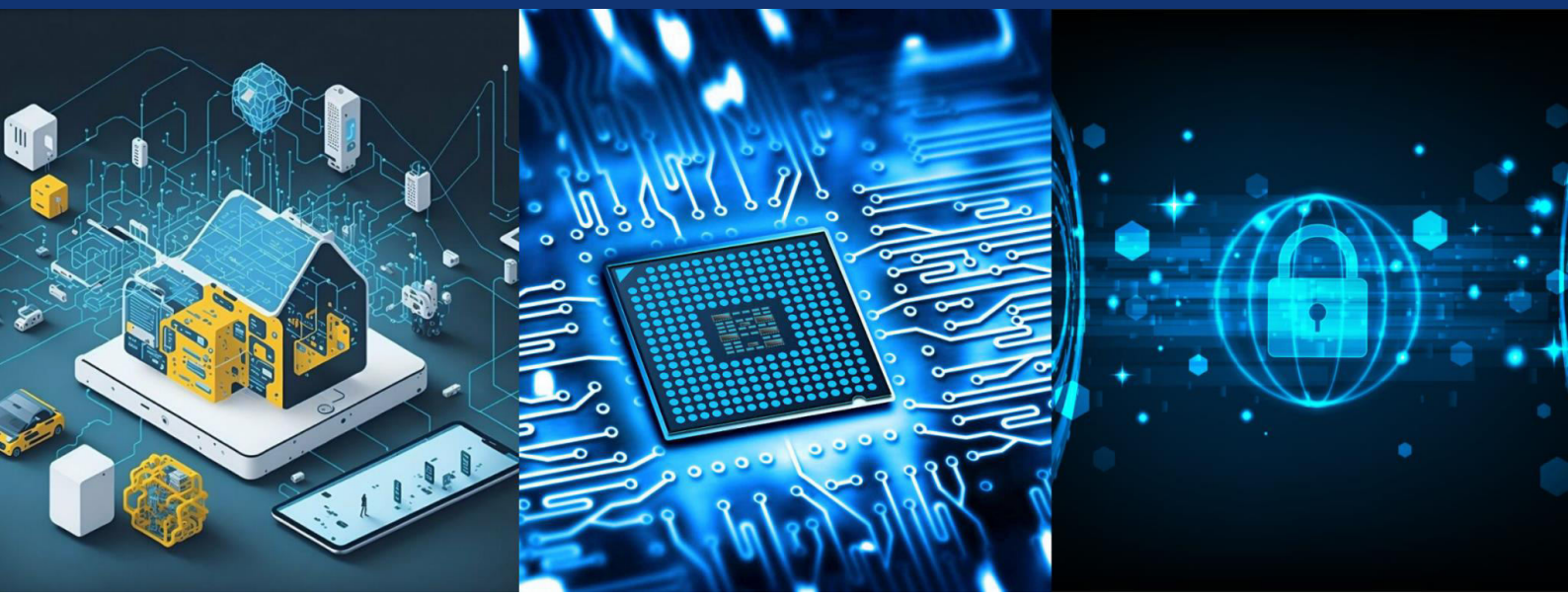
IJIRCCE

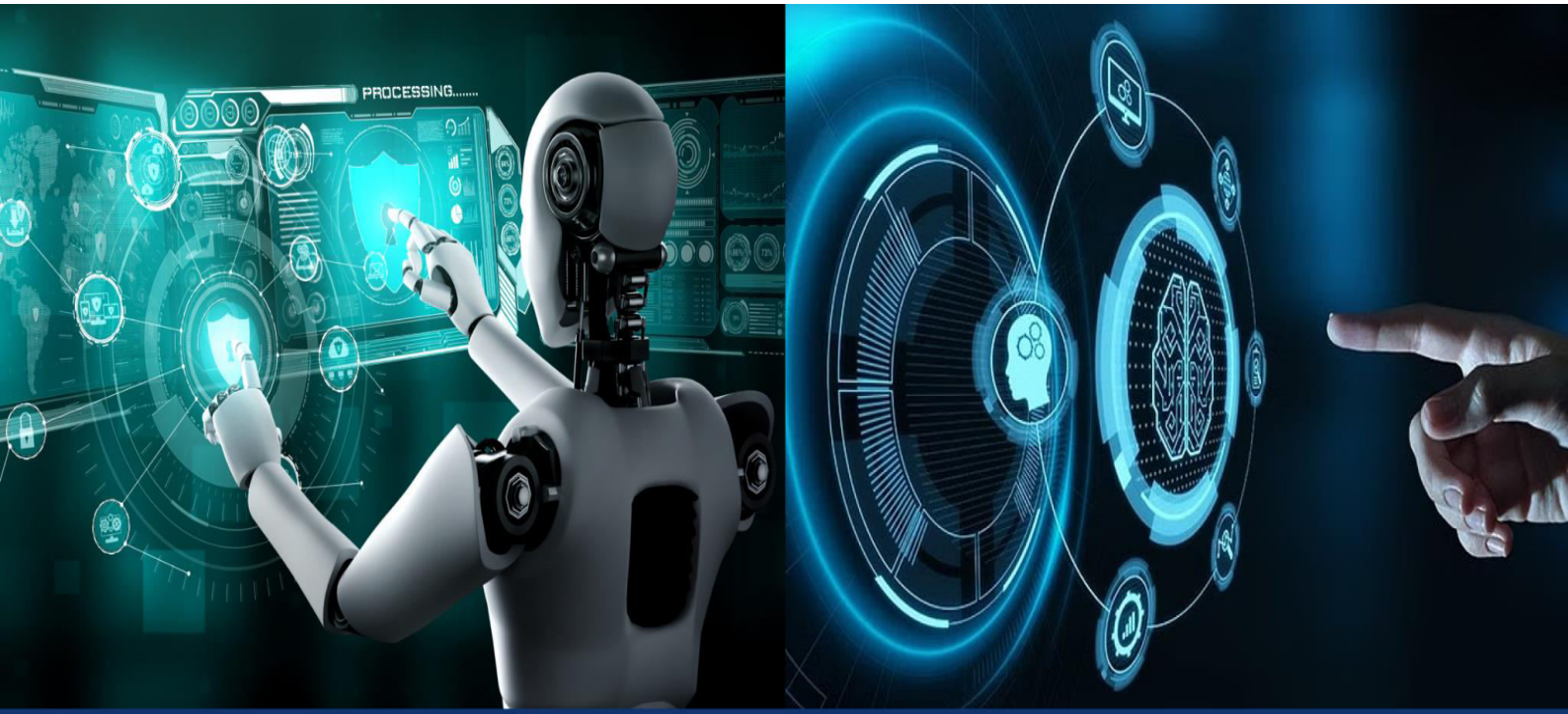
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.927

Volume 14, Issue 10, October 2026

© 2026 IJIRCCE | Volume 14, Issue 10, October 2026 DOI: 10.15680/IJIRCCE.2026.1410030

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Road Accident Prediction

Bhavana S Patil*, Pallavi K R*, Pratiksha M U*, Savita Linganagoudra*, Savitri D Baragi

Department of Computer Science and Engineering, STJ Institute of Technology, Ranebennur, Karnataka, India*

ABSTRACT: Road accidents are a major concern, and predicting their severity can help improve road safety. This research presents a Road Accident Prediction and Classification System using machine learning. The system uses historical accident data and the Random Forest algorithm to predict accident severity based on different accident-related factors. The dataset is processed before training the model to improve prediction performance. The trained model is integrated with a web application developed using Flask, HTML, CSS, and JavaScript. Users can enter the required details and view the predicted accident severity. This system aims to make accident analysis easier and support better road safety planning using machine learning.

KEYWORDS: Road Accident Prediction, Machine Learning, Random Forest, Accident Severity, Flask.

I. INTRODUCTION

Road accidents are one of the major problems affecting public safety. Predicting accident severity using machine learning can help in understanding accident patterns and improving road safety. Traditional methods of accident analysis may not efficiently identify patterns in large datasets.This research presents a Road Accident Prediction and Classification System using the Random Forest algorithm. The system uses historical road accident data to analyze different factors and predict accident severity. The trained model is integrated with a web application developed using Flask, HTML, CSS, and JavaScript. The application provides a simple interface for users to enter accident-related details and view the predicted severity. The main aim of this project is to use machine learning for accident analysis and support better road safety planning.

The main contributions of this work are as follows:

1. Data Preprocessing: Preparing historical accident data for machine learning.
2. Machine Learning Model: Using the Random Forest algorithm to classify accident severity.
3. Web Application: Integrating the trained model with Flask to provide a user-friendly interface.
4. Severity Prediction: Predicting accident severity based on the details provided by the user.
5. Road Safety Support: Helping analyze accident patterns and supporting data-driven road safety decisions.

II. RELATED WORK

Research on road accident prediction has explored machine learning algorithms and data mining techniques to analyze accident patterns and predict accident severity. Kumar et al. [1] studied road accident prediction and classification using machine learning techniques to identify patterns in accident data. Vanitha and Swedha [2] investigated different machine learning algorithms for predicting road accidents using historical data. Their work highlights the importance of analyzing accident-related factors for improving prediction. Patil and Deshmukh [3] explored traffic accident analysis and prediction using data mining techniques to extract useful information from accident datasets.These studies provide a foundation for developing machine learning-based accident prediction systems. Our project uses the Random Forest algorithm to classify accident severity based on historical accident data. The dataset is processed before training the model, and the trained model is integrated with a Flask-based web application. The system allows users to enter accident-related details and view the predicted severity. The effectiveness of the system depends on the quality of the dataset, the selected features, and the performance of the trained model. This approach aims to simplify accident severity prediction and support data-driven road safety analysis.

© 2026 IJIRCCE | Volume 14, Issue 10, October 2026 DOI: 10.15680/IJIRCCE.2026.1410030

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

III. SYSTEM ARCHITECTURE

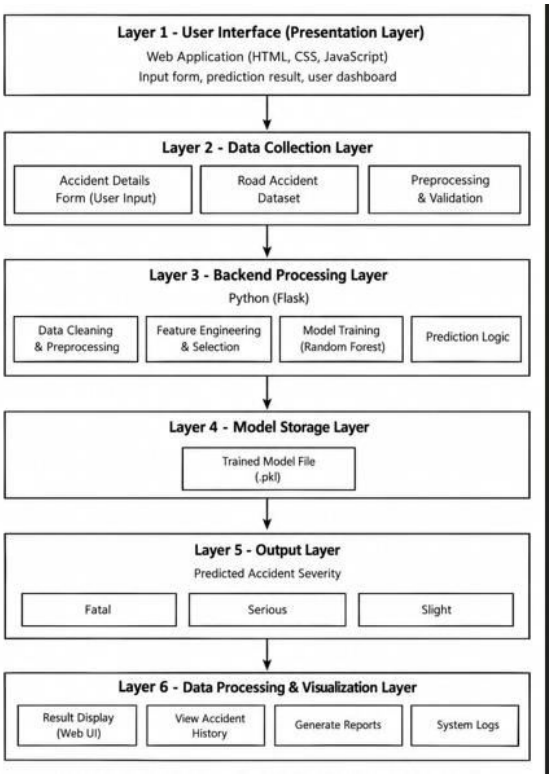


Fig.1.System architecture

1. User Layer: The user layer represents the people who interact with the system, such as traffic officers, the public, and administrators. Users provide the required accident-related details to obtain the predicted accident severity.
2. Presentation Layer: The presentation layer provides the user interface developed using Flask, HTML, CSS, and Bootstrap. It contains input forms, input validation, and a result page to display the prediction.
3. Application Layer: The application layer uses the Flask backend to process user requests. It validates the entered details, prepares the input data in the required format, and sends it to the trained machine learning model.
4. Machine Learning Layer: This layer uses the Random Forest classifier to predict accident severity. It analyzes accident-related factors such as speed, road type, weather conditions, and lighting conditions to generate a prediction.
5. Model Storage Layer: This layer stores the trained Random Forest model in a file. The saved model is loaded by the Flask application whenever a user requests a prediction, avoiding the need to train the model repeatedly.
6. Output Layer: The output layer displays the predicted accident severity to the user. The result is classified into three categories: fatal, serious, or slight, according to the model's prediction.
7. Data Processing Layer: The data processing layer prepares the historical road accident dataset for model training. It includes the following steps:

- Raw Dataset: Collects historical road accident records.
- Data Cleaning: Removes inconsistencies and prepares the data.
- Missing-Value Handling: Checks and handles missing values, if present.
- Feature Selection: Selects relevant features for predicti
- Train-Test Split: Divides the dataset into training and testing sets.
- Random Forest Training: Trains the classifier using the prepared data.
- Save Trained Model: Saves the trained model for future use in the application.

© 2026 IJIRCCE | Volume 14, Issue 10, October 2026 DOI: 10.15680/IJIRCCE.2026.1410030

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

IV. ALGORITHM

Algorithm: Road Accident Prediction Using Random Forest
Input: Accident details, including police attendance, driver age, vehicle type, vehicle age, engine capacity, day, weather, road condition, lighting condition, gender, and speed.
Output: Predicted accident severity: Fatal, Serious, or Slight.
Step 1: Dataset Loading
Load the accident, vehicle, and casualty datasets into the Python environment using Pandas.
Step 2: Data Preprocessing
Clean the data, handle missing values, remove unsuitable records, and prepare the required features for model training.
Step 3: Feature Selection
Select the relevant accident-related attributes as input features and accident severity as the target variable.
Step 4: Data Encoding
Convert categorical values into the numerical representation required by the machine-learning model.
Step 5: Dataset Splitting
Divide the prepared dataset into training and testing sets.
Step 6: Model Training
Initialize the Random Forest classifier and train it using the training dataset.
Step 7: Model Evaluation
Use the testing dataset to generate predictions and evaluate the model's performance.
Step 8: Model Saving
Save the trained Random Forest model to a file named "litemodel_new.sav" for use in the web application.
Step 9: Flask Application Initialization
Start the Flask application and load the saved model.
Step 10: User Input
Display the accident prediction form and accept the required accident details from the user.
Step 11: Input Processing
Validate the entered values and arrange the features in the same order used during model training.
Step 12: Severity Prediction
Pass the prepared input to the trained Random Forest model and obtain the predicted severity class.
Step 13: Result Generation
Interpret the predicted class as follows:
- Class 1 – Fatal Accident
- Class 2 – Serious Accident
- Class 3 – Slight Accident

Step 14: Result Display
Display the predicted accident severity on the web page.
Step 15: Additional Features
Allow users to access accident visualizations or the configured SMS demonstration, where available.
Step 16: Repeat or Stop
Allow the user to submit another prediction while the application is running. Stop the Flask server when the application is no longer required.

V. IMPLEMENTATION DETAILS

The Road Accident Prediction and Classification System is implemented using Python, Flask, and machine learning techniques. The Random Forest classifier is used to predict accident severity based on historical accident data and user-provided details. The implementation begins by loading the accident, vehicle, and casualty datasets using Pandas. The collected data is then cleaned, missing values are handled, and unsuitable records are removed where necessary. Relevant features such as driver age, vehicle type, vehicle age, engine capacity, weather conditions, road conditions, lighting conditions, and speed are selected for prediction. Categorical values are converted into numerical representations so that they can be processed by the machine learning model.

© 2026 IJIRCCE | Volume 14, Issue 10, October 2026 DOI: 10.15680/IJIRCCE.2026.1410030

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

The prepared dataset is divided into training and testing sets. The Random Forest classifier is trained using the training data to learn patterns associated with accident severity. The trained model is evaluated using the testing data and suitable performance metrics. After training, the model is saved in a file named litemodel_new.sav for future use.
The Flask application loads the saved model and provides a web interface for users to enter accident-related details. The entered values are validated and arranged in the same format used during training. The Random Forest model processes the input and predicts the accident severity. The result is displayed on the web page as Fatal, Serious, or Slight, based on the model's class mapping. The application may also provide accident visualizations and an SMS demonstration where these features are implemented. Users can submit multiple predictions while the application is running.

VI. EXPERIMENTAL EVALUATION

The Road Accident Prediction and Classification System is evaluated using historical road accident data containing accident-related information. The dataset is preprocessed by cleaning the data, handling missing values where necessary, selecting relevant features, and converting categorical values into numerical representations. The prepared dataset is divided into training and testing sets to train and evaluate the Random Forest classifier.

A. Experimental Setup: The system is developed using Python, Pandas, Scikit-learn, and Flask. The Random Forest model is trained using the prepared accident dataset and integrated with the web application. Functional testing checks whether users can enter accident details, submit the form, and receive the predicted accident severity.

B. Evaluation Metrics: The model's performance can be measured using accuracy, precision, recall, and F1-score. A confusion matrix can also be used to examine the classification of fatal, serious, and slight accidents. The actual metric values should be included after testing the model on the prepared test dataset.

C. Observations: The system combines machine learning with a web interface to make accident severity prediction easier. The quality of predictions depends on the dataset, selected features, and model performance. The system supports accident analysis, but its predictions should not be considered a guarantee of actual accident outcomes.

Module	Description
Dataset Preparation	Loads and preprocesses historical road accident data.
Feature Selection	Selects relevant accident-related features for prediction.
Data Splitting	Divides the dataset into training and testing sets.
Random Forest Training	Trains the classifier using the training dataset.
Model Evaluation	Evaluates prediction performance using accuracy, precision, recall, and F1-score.
Severity Classification	Classifies accidents as fatal, serious, or slight.
Flask Application	Accepts user inputs and connects the saved model to the web interface.
Prediction Results	Displays the predicted accident severity to the user.

VII. DISCUSSION

The proposed Road Accident Prediction and Classification System uses the Random Forest algorithm to classify accident severity based on historical data and user-provided information. The model analyzes relevant factors such as driver age, vehicle type, vehicle age, weather, road conditions, lighting conditions, and speed, depending on the available input features.

The Flask-based web application allows users to enter accident details and view the predicted severity as fatal, serious, or slight. Saving the trained model in a file allows the application to reuse it without retraining for every prediction. The system provides a simple way to apply machine learning to road accident analysis. However, the reliability of the predictions depends on the quality and representativeness of the dataset, correct input processing, and the model's evaluation results. Therefore, the system should be considered a decision-support tool rather than a replacement for professional road safety analysis.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

VIII. FUTURE WORK

Future development of the Road Accident Prediction and Classification System may include using larger and more recent accident datasets to improve the model's ability to recognize different accident patterns. Additional features, such as traffic density, road geometry, vehicle condition, and geographical location, could be included where reliable data is available.

The system could also be enhanced with interactive maps to identify accident-prone areas and display accident patterns across different locations. Comparing Random Forest with other machine learning algorithms may help identify a more suitable model. Further work could include improving the web interface, providing visual reports, and integrating real-time traffic and weather information where available. Future evaluation should test the model on independent datasets and examine its performance across all severity classes to improve reliability and usefulness for road safety planning.

IX. CONCLUSION

The Road Accident Prediction Using Machine Learning project aims to predict road accident severity using accident-related data and the Random Forest algorithm. The system processes relevant details, such as driver age, vehicle type, weather, road conditions, and speed, to classify accidents as fatal, serious, or slight. The trained model is integrated into a Flask web application, allowing users to enter accident details and view the predicted severity. This project demonstrates the application of machine learning in road safety analysis and helps identify factors that may influence accident severity. Although the system provides useful predictions, its reliability depends on the quality and quantity of the available data. In the future, the project can be improved by using larger datasets, enhancing model performance, and adding more accident visualizations to support better road safety planning.

REFERENCES

[1] K. S. Kumar, A. K. S. Reddy, and M. Prakash, "Road Accident Prediction and Classification Using Machine Learning Techniques," International Journal of Innovative Technology and Exploring Engineering (IJITEE), 2022.
[2] R. Vanitha and M. Swedha, "Prediction of Road Accidents Using Machine Learning Algorithms," Middle East Journal of Applied Science & Technology (MEJAST), vol. 6, по. 2, pp. 64-75, May 2023.
[3] S. R. Patil and M. S. Deshmukh, "Traffic Accident Analysis and Prediction Using Data Mining Techniques," International Journal of Advanced Research in Computer Science, 2021.



INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details