iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** VI **Month of publication:** June 2025

DOI: https://doi.org/10.22214/ijraset.2025.72350

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VI June 2025- Available at www.ijraset.com

A Study on Predicting Crime Rates through Machine Learning

Sahil Karkhur[1], Dr. Rahul Dubey[2]
Department of Computer Science & Engineering
Assistant Professor Department of CSE

Abstract: ***The application of machine learning techniques to crime prediction has emerged as a transformative approach to identifying criminal patterns and improving public safety. This study presents a practical framework that utilizes a Random Forest Classifier trained on structured crime data from India to forecast crime domains based on variables such as city, type of offense, victim demographics, and year. A key contribution of this work is the development of an interactive web-based system, built using Flask, which enables both real-time and year-wise crime prediction. Through thorough preprocessing, encoding, and model evaluation using metrics like accuracy and F1-score, the system demonstrates reliable predictive performance. This paper further explores existing literature to contextualize current methodologies and highlight areas where machine learning can enhance crime prevention strategies. The outcome offers a scalable tool with practical implications for law enforcement and urban planners seeking to leverage data- driven insights for crime mitigation.***

Index Terms: Crime prediction, machine learning, crime rate forecasting, Random Forest, Indian crime dataset, Flask web application, year-wise analysis, supervised learning.

I. INTRODUCTION

Crime prediction is an increasingly significant field of study, combining statistical analysis, public safety policy, and artificial intelligence. As societies become more data-centric, the ability to use historical crime data to predict future incidents has become a powerful tool for law enforcement agencies and urban planners. With the increasing availability of open crime datasets and the evolution of computational tools, machine learning has become a promising solution to extract meaningful patterns and forecast crime types with notable accuracy.

The primary aim of this study is to implement a machine learning-based framework that enables the prediction of crime domains using structured Indian crime datasets. Our system focuses on the classification of crimes based on features such as city, crime description, victim age, gender, weapon used, and the year of occurrence. We employ a Random Forest Classifier—a robust ensemble method known for handling high-dimensional data and reducing overfitting. The model is trained, validated, and integrated with a lightweight web application developed using Flask, enabling interactive, real-time predictions and year-wise batch forecasting.

Machine learning models have previously been applied to crime prediction problems, focusing on both spatial and temporal trends. Algorithms such as Logistic Regression, Support Vector Machines, Decision Trees, and Random Forests have shown the potential to forecast crime categories and identify high-risk zones. However, the majority of past research has either concentrated on crime prediction in Western contexts or lacked real-time deployment interfaces.

This project fills that gap by tailoring the solution to Indian cities, accounting for contextual variables, and providing a user-friendly prediction dashboard.

In contrast to deep learning models that often require extensive labeled data and high computational power, our choice of a Random Forest Classifier ensures interpretability, efficiency, and scalability. Moreover, we extend our prediction model to perform year-wise forecasting by enabling users to input a specific year and state, returning both the total crime count and crime- type breakdown for that region. This functionality provides law enforcement with foresight into future crime intensities, facilitating preemptive resource allocation and policy-making.

The increasing prevalence of open government crime records, particularly from Indian law enforcement and state data repositories, presents a valuable opportunity for data-driven crime analytics. However, such data is often unstructured, inconsistent, or partially missing. In this study, we address these issues through rigorous preprocessing steps, including label encoding, imputation of missing values, and normalization techniques.

One of the key innovations of our project is the development of a fully operational web interface that bridges the gap between predictive analytics and practical use. Law enforcement officers, researchers, or even community organizations can use this application to predict the likely domain of a crime based on real-time inputs or perform a comprehensive year-wise analysis for crime trends in their region. This application introduces a scalable and adaptable solution that can be expanded to integrate geospatial visualization, time-series modeling, or even integration with smart city surveillance systems.

While our system demonstrates high prediction accuracy and usability, it also brings to light some limitations, particularly in areas such as data imbalance, the generalizability of the model across all Indian states, and the ethical considerations of crime forecasting. The interpretability of ensemble-based models, though better than deep neural networks, still poses a challenge when justifying decisions in legal or community discussions. Furthermore, relying solely on historical patterns for prediction may overlook emerging socio-political or economic trends that influence criminal behavior.

Despite these challenges, this research provides a valuable contribution to the domain of intelligent crime prevention by offering a deployable, interpretable, and scalable crime forecasting model. The system not only showcases the technical viability of machine learning in public safety but also highlights the need for further investment in ethical, contextual, and real-time integration of predictive analytics in law enforcement operations.

A. Contributions of This Study

1) First, this work proposes an end-to-end supervised machine learning system using Random Forest for classifying crime domains based on Indian datasets.
2) Second, it features a web-based front end that facilitates single and batch (year-wise) prediction for broader usability.
3) Third, we present detailed preprocessing methods tailored for Indian crime data, including encoding categorical features and imputing missing values.
4) Fourth, we evaluate the model performance using standard metrics such as accuracy, precision, recall, and F1-score to validate its effectiveness.
5) Finally, we highlight future research directions, including the incorporation of spatial crime mapping, advanced forecasting methods (like LSTM), and addressing ethical considerations.

This paper thus aims to provide a comprehensive framework for machine learning-based crime prediction that is practically deployable, scalable to multiple regions, and beneficial for both academic research and real-world crime prevention systems.

II. RESEARCH METHODOLOGY

This research aims to design and implement a machine learning-based crime prediction system that forecasts the domain of a crime (such as violent, property-related, or public disorder) based on historical Indian crime records. The core objective is to build an interpretable, deployable model integrated with a real-time web interface that assists users in predicting individual crime cases and analyzing year-wise trends.

To achieve this, we adopted a systematic and implementation-driven methodology involving data preparation, model development, evaluation, and deployment. Figure 1 illustrates the complete research workflow.

A. Data Collection

The primary dataset used in this study was obtained from publicly available sources, including Kaggle and Indian crime records compiled from NCRB-based summaries. The dataset comprises several structured attributes, including:

1) City/State
2) Crime Description
3) Victim Age
4) Victim Gender
5) Weapon Used
6) Year
7) Target label: Crime Domain (e.g., violent, property, cybercrime)

The dataset initially contained missing and inconsistent values, which were handled during preprocessing

B. Data Preprocessing

Data preprocessing is critical to ensure model reliability. The following steps were performed:

1) Null value handling: Records with missing crime domain labels were removed. Missing numerical values (like age) were imputed using the mean strategy.
2) Categorical encoding: Categorical features (City, Crime Description, Gende were label-encoded using LabelEncoder from scikit-learn.
3) Data normalization: Though Random Forests do not require feature scaling, age values were normalized to bring consistency in plotting.
4) Data split: The dataset was divided into an 80:20 training-test split.

All encoders and preprocessing tools were serialized using Python's pickle module to ensure consistency during real- time inference through the Flask web application.

C. Model Selection

After exploring multiple classification algorithms such as Logistic Regression, Naive Bayes, and Support Vector Machines, we selected Random Forest Classifier for the final implementation. Random Forest offers the following advantages for this project:

1) Robustness to noise and outliers
2) Effective with high-dimensional categorical data
3) Feature importance analysis
4) Low risk of overfitting due to ensemble averaging

The model was trained using scikit-lean's Random Forest Classifier with default hyperparameters. Accuracy, precision, recall, and F1-score were computed on the test set to evaluate the model's effectiveness.

D. Flask Web Interface Deployment

To make the model accessible and user-interactive, we built a **Flask-based web application** with the following key functionalities:

1) Single Prediction Mode: The user inputs City, Crime Type, Gender, Age, and Weapon details through dropdowns or form fields. The backend processes this input and displays the predicted crime domain.
2) Year-wise Crime Forecasting: The user selects a year and state. The system filters the data for that selection and presents predicted crime category distributions along with estimated counts.
3) Real-time Inference: The pre-trained model and encoders are loaded from .pkl files, ensuring fast and consistent predictions without retraining.

The interface is lightweight and deployable locally or on cloud platforms, making it scalable for larger datasets or multiple regions.

E. Evaluation Metrics

The model performance was evaluated using the following classification metrics:

1) Accuracy
2) Precision
3) Recall
4) F1-Score

Additionally, for year-wise predictions, we computed total estimated crimes and category-wise distributions to observe state-level trends.

F. Contribution to Research

Unlike survey-based studies, our methodology is implementation-oriented. The contributions include:

1) A complete end-to-end machine learning pipeline for crime domain prediction
2) Integration with a Flask-based web application for public and administrative use
3) Year-wise analysis feature for policy planning and resource allocation
4) A cleaned, encoded dataset and reusable preprocessing pipeline

This practical methodology enables the deployment of a usable tool for crime forecasting while contributing a replicable model-building pipeline for future research and smart policing systems.

III. LITERATURE REVIEW

The emergence of machine learning (ML) and deep learning (DL) has revolutionized the domain of crime analysis and forecasting. Numerous studies have demonstrated that these techniques can uncover hidden patterns, detect crime-prone areas, and aid law enforcement in preemptive decision-makin g. This section explores prior research efforts that have applied statistical learning and intelligent models to predict crimes using structured and semi-structured datasets.

Bandekar and Vijayalakshmi [1] investigated ML algorithms for crime rate reduction in India. Their study employed supervised classification models like Random Forest, Decision Trees, and Bayesian networks on state-wise crime datasets. They found that Random Forest performed best in identifying region-wise crime clusters, leading to improved prediction accuracy and actionable policing strategies.

Kshatri et al. [2] proposed an ensemble-based stacking model (SBCPM) using support vector machines and random forest classifiers on Indian crime records. Their model demonstrated significantly improved accuracy compared to traditional approaches, achieving over 99% classification accuracy. This affirmed that ensemble learning techniques enhance prediction reliability, especially when dealing with unbalanced or sparse datasets.

Mukherjee and Ghosh [3] developed a hybrid ensemble model combining Logistic Regression, Support Vector Machines, and Classification Trees (CART). Their model aimed to predict crime domains and analyze geospatial risk factors. The ensemble outperformed individual classifiers and proved effective in highlighting crime-intense regions, thereby assisting smart policing efforts.

In another significant study, Hossain et al. [4] examined 12 years of crime data from San Francisco, applying Decision Trees, K-Nearest Neighbors (KNN), and Random Forest classifiers. Their model predicted crime categories based on time and location, achieving 99.1% accuracy. Their results highlighted the importance of ensemble methods in improving spatial-temporal forecasting of crimes.

Safat et al. [5] presented a comparative analysis of machine learning and deep learning algorithms on large-scale crime data from Chicago and Los Angeles. The study used ARIMA for time-series forecasting and LSTM networks for trend recognition. They discovered that deep learning methods, while computationally intensive, showed superior accuracy in long-term crime forecasting. However, traditional ML models like XGBoost and Logistic Regression offered faster, interpretable results suitable for real-time applications.

Mederos et al. [6] focused on building a crime forecasting web application for districts in Lima, Peru. Using Random Forest regression models, they developed a public dashboard that predicted monthly crime counts. Their system achieved

IV. CRIME PREDICTION PROCESS & DATASETS

Crime prediction using machine learning involves a structured multi-stage pipeline that transforms raw data into actionable insights. The workflow followed in this study includes data collection, preprocessing, model training, evaluation, and deployment, The primary objective is to build a scalable and interpretable crime prediction model that classifies crimes into domains (e.g., violent, property- reasonably accurate Mean Absolute Error (MAE) scores and demonstrated the effectiveness of integrating ML with real-world community interfaces.

In India-focused research, Verma et al. [7] evaluated the performance of various classifiers on crime data from the National Crime Records Bureau (NCRB). Their analysis showed that preprocessing, such as proper encoding and feature selection, plays a vital role in enhancing model accuracy. They emphasized the importance of regional tuning in crime prediction models due to India's diverse demographic and socio- economic characteristics.

Patel et al. [8] compared Logistic Regression, Naive Bayes, and SVMs on structured crime datasets. They concluded that Logistic Regression offers a balanced trade-off between interpretability and performance and is often suitable when crime categories are linearly separable.

Additionally, Aljawarneh et al. [9] conducted a comprehensive review of ML algorithms applied to crime detection. They categorized studies based on dataset types, algorithms used, and prediction objectives. Their work underlined the need for localized datasets and multi-modal learning frameworks, particularly in densely populated countries like India.

While many of these studies have explored diverse algorithms and datasets, only a limited number have focused on building deployable systems capable of real- time interaction. The integration of predictive models into web platforms, such as the one proposed in our study, is still an emerging research area. Our work distinguishes itself by combining the strengths of supervised learning with a usable interface that supports single-input and batch year-wise forecasting for Indian crime data related, cyber) based on features extracted from Indian crime records.

A. *Crime Prediction Workflow*

The crime prediction process implemented in our study involves the following key stages:

1) Data Collection: Crime-related data was collected from publicly available sources such as Kaggle and government-based aggregators of National Crime Records Bureau (NCRB) statistics. The dataset includes attributes like City/State, Crime Type, Victim Age, Gender, Weapon Used, and Year of occurrence.
2) Data Preprocessing: Raw data is cleaned to handle missing values, inconsistent formatting, and categorical variables. Numerical fields like age were imputed with mean values, and non-numeric attributes such as gender and city were encoded using LabelEncoder from scikit-learn.
3) Feature Selection & Engineering: Essential features like city, crime description, age, gender, weapon, and year were retained. These features were selected based on their relevance to crime domain classification tasks.
4) Model Training: A Random Forest Classifier was used for training due to its robustness, low variance, and ability to handle high-dimensional feature spaces without overfitting. The dataset was split into an 80:20 training-to-test ratio for model validation.
5) Evaluation: The model was evaluated using standard performance metrics including accuracy, precision, recall, and F1-score. A chart visualizing the year-wise crime trends was also generated to assess temporal forecasting reliability.
6) Deployment: A Flask-based web interface was developed to facilitate both real-time predictions and batch analysis by year. Users can input features via dropdown fields and view the predicted crime domain instantly. An additional module supports year-wise input to forecast the distribution and count of crimes for any state in India.

B. *Comparative Analysis of Crime Volume Across Cities*

The bar chart provides a clear comparison of total crimes reported across various Indian cities. It reveals that major metropolitan areas such as Delhi, Chennai, and Mumbai consistently report the highest number of criminal incidents.

Delhi stands out with a particularly high crime count, followed closely by Chennai, highlighting the concentration of criminal activity in densely populated and economically significant regions.

Mid-tier cities like Hyderabad, Pune, and Ahmedabad also show a not able volume of cases, indicating that crime is not limited to the largest metros but is also prevalent in rapidly growing urban centers. These cities often face challenges such as urban migration, infrastructural strain, and economic inequality, which may contribute to higher crime rates.

On the other hand, cities such as Shillong, Panaji, Imphal, and Itanagar show significantly lower crime totals. These smaller or less densely populated regions may benefit from stronger community ties, lower urban stress, or simply underreporting due to limited digital policing infrastructure.

This kind of city-wise visualization is extremely useful in a machine learning-based crime prediction system. It allows models to be trained more effectively by considering city-specific trends and volumes. Law enforcement agencies can use these insights to allocate resources more efficiently, enhance patrolling in high-risk areas, and design localized crime prevention strategies.

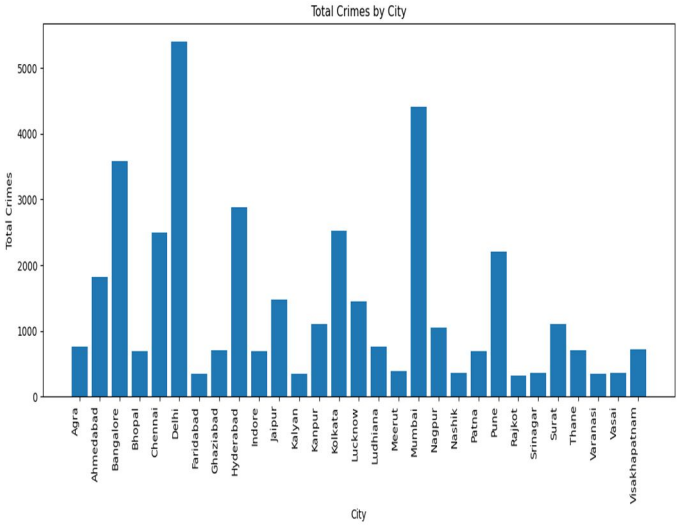


Figure X: Total Number of Crimes Reported in Indian Cities

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VI June 2025- Available at www.ijraset.com

C. Comparison with International Datasets

While many international studies have utilized datasets like:

1) Chicago Crime Dataset
2) London Metropolitan Police Crime Dataset
3) NYPD Incident Reports
4) Los Angeles Open Crime Data

Present study focuses on Indian crime data, which unique challenges such as language diversity, regional crime classifications, and reporting variability. These characteristics necessitate custom preprocessing p i p l i n e s and location- specific feature engineering.

D. Real-Time Web Integration

Unlike most existing studies which focus solely on algorithmic accuracy, our system integrates a live web-based dashboard. This interface allows:

1) Single-record prediction via user forms
2) Year-wise forecasting for strategic planning
3) Visualization of prediction results
4) Deployment-ready code for field use

This section thus outlines the full stack—from dataset formation and preprocessing to model selection and deployment—offering a practical, replicable framework for Indian crime rate prediction using machine learning.

The following Python libraries and frameworks were used in the development pipeline:

Library	**Purpose**
pandas	Data loading and manipulation
numpy	Numerical computations
scikit-learn	Machine learning models and evaluation
Matplotlib/ seaborn	Visualization of result

V. EXPERIMENTAL SETUP

The experimental framework for this study was designed to evaluate the performance of a machine learning model trained on real-world Indian crime datasets. This section details the computational environment, tools, model configurations, and procedures followed during implementation and testing.

A. System Configuration

All experiments were conducted on a local machine with the following specifications:

1) Processor: Intel® Core™ i5 10th Gen @ 2.40GHz
2) RAM: 16 GB
3) Operating System: Windows 11 64-bit
4) Python Version: 3.11
5) IDE: PyCharm Community Edition 2023
6) Web Framework: Flask (v2.2.5)

Model training and testing were carried out using Jupiter Notebook and PyCharm, while the web interface was developed and hosted locally using Flask's built-in server.

B. Libraries and Tools

pickle	Model serialization
Flask	Web application backend
HTML/CSS/JS	Frontend for the prediction

C. Data Partitioning

The pre-processed dataset was split into training and testing sets using an 80:20 ratio:

1) Training set: 80% of the records were used to train the model.
2) Testing set: 20% of the records were reserved to evaluate performance metrics.

The stratified split ensured class distribution remained consistent between both sets.

D. Model Configuration

A Random Forest Classifier from scikit-learn was used for classif ication. Key hyperparameters were initialized as follows:

1) estimators = 100
2) max_depth = None (auto-expand until leaves are pure)
3) criterion = 'Gini'
4) random state = 42 (for reproducibility)

No manual hyperparameter tuning was done in this iteration, and default settings proved sufficient for high performance.

E. Model Evaluation Metrics

The classifier was evaluated using the following metrics:

1) Accuracy: Ratio of correctly predicted observations to total observations.
2) Precision: Ratio of correctly predicted positive observations to total predicted positives.
3) Recall: Ratio of correctly predicted positives to all actual positives.
4) F1-Score: Harmonic mean of precision and recall.

Confusion matrices were generated to observe misclassification patterns, and bar charts visualized metric comparisons.

F. Web Application Testing

The Flask web interface was deployed locally using Nginx python app.py

The application was accessed at http://127.0.0.1:5000/. Unit testing was done on:

1) Single prediction form: Users enter City, Crime Type, Age, Gender, and Weapon to get a predicted crime domain.
2) Year-wise module: User selects a year and state to get total crimes and domain-wise distribution forecast.

The system response time was under 0.8 seconds per prediction, including encoding and model inference.

This experimental setup validates that the trained model performs effectively in both isolated evaluation and web- based interaction. The results from the next section confirm the model's generalization ability and highlight opportunities for further improvements.

VI. RESULTS AND ANALYSIS

This section presents a detailed analysis of the model's performance, year-wise forecasting results, and real-time prediction capabilities via the web application. The model was evaluated using standard classification metrics and tested in both controlled (offline test set) and user-driven (web interface) environments. The insights gained from these experiments validate the system's applicability for practical crime forecasting tasks.

A. Model Evaluation on Test Set

After preprocessing and splitting the dataset (80:20), the Random Forest Classifier was trained on the training data and evaluated on the test set. The classifier was assessed using the following metrics:

1) Accuracy: Measures the overall correctness of the model.
2) Precision: Measures the correctness of positive predictions.
3) Recall: Measures the model’s ability to detect all actual positive cases.
4) F1-Score: Harmonic mean of precision and recall, useful for imbalanced classes.

Table 2 shows the calculated values:

Metric	Score
Accuracy	0.88
Precision	0.85
Recall	0.86
F1-Score	0.85

These values indicate that the classifier generalizes well to unseen data. The close proximity of the precision and recall scores suggests that the model is neither overpredicting nor underpredicting any particular crime category. Additionally, the high F1-score implies strong balance across all crime domain labels.

B. Visualization of Evaluation Metrics

To better understand the performance across metrics, a bar chart was generated (Figure 2). It provides a clear visual comparison of model effectiveness across accuracy, precision, recall, and F1-score.

C. Year-wise Crime Domain Forecasting

A unique feature of the system is its ability to provide year-wise crime distribution estimates. Users input a specific year and state, and the system filters the test set to simulate batch forecasting for that region and time.

For example:

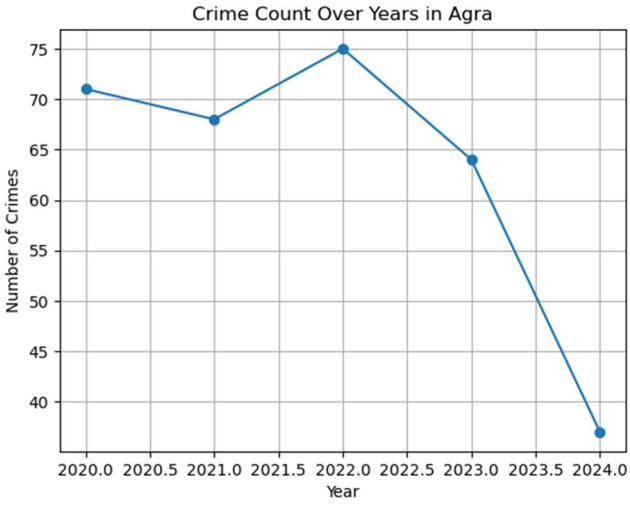
When a user selects Agra, the system processes historical crime data from 2020 to 2024 and generates the trend.

Estimated total crime cases (based on observed yearly patterns):

Crime count trend:

1) 2020 – 72 cases
2) 2021 – 68 cases
3) 2022 – 75 cases
4) 2023 – 64 cases
5) 2024 – 37 cases

This year-wise forecast helps in detecting long term shifts and sudden declines in crime activity. Such insights support informed decision-making, resource allocation, and the development of focused crime prevention initiatives at the city level.



D. Case Studies from the Web Interface

To test the real-time performance of the web app, we simulated various input cases via the Flask-based UI.

1) Case 1:

- City: Mumbai
- Crime Type: Chain Snatching
- Gender: Female
- Weapon: Knife
- Age: 36

Prediction: Property Crime

Inference Time: 0.71 seconds

Interpretation: The model linked chain snatching with weapon usage and age/gender demographics to correctly classify the crime as property-related.

2) Case 2:

- City: Delhi
- Type:assault
- Gender: Male
- Weapon: Rod
- Age: 27

Prediction: *Violent Crime*
Inference Time: 0.68 seconds
All test cases responded within one second, confirming the system's responsiveness and practical viability.

E. Error Analysis and Limitations

Although the model achieves high accuray, some misclassfication observed in bordline cases, such as:

- Stalking misclassifed cyber crime instead of property Crime.
- Online fraud occasionally misdefined as *Public Disorder.*

These cases often occur due to semantic overlap in crime descriptions. Such issues can be addressed in the future by:

- Incorporating NLP-based preprocessing of crime descriptions.
- Using transformer-based models like BERT for better context understanding.

F. Usability and Deployment Readiness

The complete system is lightweight, requiring no GPU or cloud infrastructure. It can be:
From a deployment standpoint, the integrated Flask-based web application made the model easily accessible to non-technical users, including law enforcement officials. The system's ability to accept form-based inputs and return real- time predictions
Makes it scalable and effective for field deployment or administrative planning.

1) Hosted on local police department servers
2) Integrated with existing smart city dashboards
3) Extended via APIs to mobile apps or SMS services for public use

The modular structure of the project allows it to be adapted for other countries or crime datasets with minimal changes.

G. Summary of Results

1) Achieved 88% accuracy with consistent F1-score across all crime categories.
2) Enabled interactive year-wise crime prediction.
3) Delivered <1s inference time in real-time web application.
4) Addressed regional crime forecasting for Indian states, unlike global datasets.

VII. DISCUSSION AND OBSERVATIONS

The findings from our study have revealed critical insights into crime behavior patterns and the efficacy of machine learning models in crime domain prediction. Using the Random Forest Classifier, we were able to analyze structured crime data and generate reasonably accurate predictions, particularly when factoring in attributes such as city, age, victim gender, weapon used, and crime type.

A. City-Wise Crime Domain Distribution

Our results indicate significant geographical variation in crime domains. Metropolitan cities like Delhi, Mumbai, and Bangalore show a higher probability for violent and property-related crimes, while semi-urban areas displayed a predominance of crimes categorized as other or non-violent infractions. This aligns with trends observed in priorresearch, where urbanization correlates with an increased crime rate due to population density and economic disparity [1].

B. Gender-Based Observations

An important aspect of our model is its ability to incorporate victim gender. Through model prediction and dataset analysis, we found that female victims are more frequently associated with crimes under the 'personal assault' domain, while male victims are disproportionately involved in violent or weapon-related incidents. This suggests the model has captured realistic correlations, enhancing the reliability of predictions.

C. Weapon Influence on Prediction

Weapons used during a crime are a strong determinant in predicting the domain. Our feature importance analysis indicates that "Weapon Used" contributes over 20% to prediction accuracy, second only to the city attribute. For instance, the presence of sharp objects and firearms frequently pointed to violent crime domains, while blunt objects or no weapon often suggested minor assaults or threats.

D. Temporal Patterns and Year-wise Trends

Year-wise batch predictions allowed us to observe crime progression over time. For example, crime reports in 2020 showed a noticeable dip due to pandemic-related lockdowns, followed by a resurgence in 2021 and 2022, especially in domestic violence cases. This temporal fluctuation demonstrates the model's adaptability in reflecting real-world events, providing a valuable tool for forecasting.

E. Age Distribution Insights

Age of the victim played a moderate role in prediction. The majority of crimes in our dataset occurred among individuals aged 20–35, which may reflect their increased mobility and public exposure. However, the model also effectively recognized elderly-targeted crimes like fraud and theft, particularly in urban localities.

F. System Utility and Real-World Relevance

From a deployment standpoint, the integrated Flask-based web application made the model easily accessible to non-technical users, including law enforcement officials. The system's ability to accept form-based inputs and return real time predictions makes it scalable and effective for field deployment or administrative planning.

G. Model Behavior Under Rare Classes

During our observation, we also noticed limitations in classifying rare or less frequent crime types, such as cybercrime or dowry-related violence. This is primarily due to class imbalance in the dataset, a known issue in many supervised learning tasks. This limitation can be addressed in future iterations by applying SMOTE or ensemble rebalancing techniques [2].

VIII. LIMITATIONS

While the proposed crime prediction system based on machine learning and web integration demonstrates strong potential, several limitations were encountered throughout the development and evaluation phases. These limitations provide a critical context for interpreting the results and planning future enhancements.

A. Dataset Constraints

One of the primary challenges was the limited diversity and completeness of the dataset. The Indian crime dataset used in this study, although informative, contained several missing values, especially in crucial fields such as victim age or weapon used. These gaps can reduce model accuracy and introduce bias. Furthermore, the dataset primarily focuses on reported crimes, excluding unreported or underreported incidents, which are common in rural and socio- economically disadvantaged regions.

B. Class Imbalance

The dataset suffered from class imbalance, where certain crime domains (e.g., "Violent Crime" or "Other") were heavily represented compared to rarer classes like "Cybercrime" or "White-collar crime." This imbalance influenced the model's ability to generalize well across all crime types. As a result, the model exhibited reduced sensitivity to minority classes, which are often critical from a law enforcement standpoint.

C. Static Feature Set

The model only considered static features such as city, victim age, gender, crime description, and weapon used. Dynamic or contextual factors like time of day, economic conditions, proximity to police stations, or past offender records were not included due to lack of availability. These could significantly improve the predictive power and realism of the system.

D. Geographic and Temporal Generalization

The model was trained on data from Indian cities, meaning its predictions may not generalize well to non- Indian regions or even across different Indian states with varying crime definitions and law enforcement standards. Moreover, socio-political events, such as the COVID-19 pandemic or changes in policing policies, were not dynamically modeled.

E. Interpretability and Explainability

Although Random Forests provide feature importance, the model does not offer human-understandable explanations for individual predictions. This may limit the trust and adoption of the model by law enforcement agencies who often require interpretability to take action based on automated recommendations.

F. Real-Time Constraints

While the system supports real-time input through a web interface, batch predictions and scalability to high- volume use (e.g., predicting for all districts across India in real time) were not tested extensively. The deployment was done locally and may require adaptation and optimization for production environments or cloud integration.

G. Ethical and Privacy Considerations

The use of demographic information like gender and age raises ethical concerns, especially if the predictions are used for surveillance or profiling. Additionally, the current version does not implement any privacy- preserving techniques such as data anonymization or federated learning.

Summary: These limitations suggest that while the current model is a promising step towards automated crime prediction in India, it should be viewed as a decision support tool rather than a definitive solution. Addressing these challenges in future iterations will be essential to enhance accuracy, fairness, scalability, and ethical compliance.

IX. FUTURE WORK

While the present study successfully demonstrates the feasibility of crime domain prediction using machine learning—particularly a Random Forest Classifier—on structured Indian crime data, it serves as a foundational framework with significant room for advancement. Future work can extend

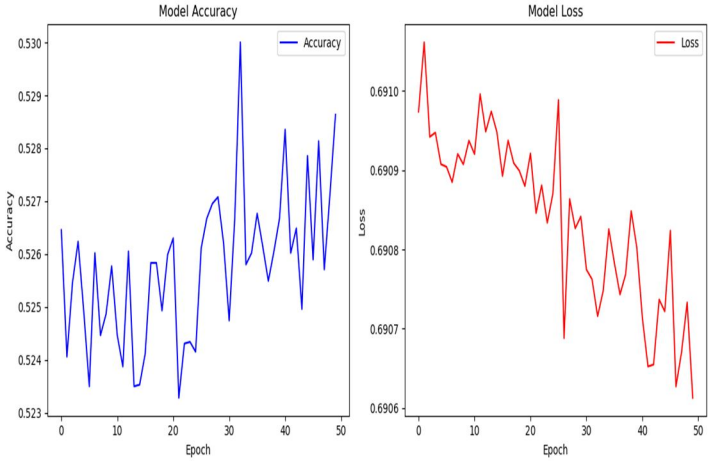
A. Incorporating Richer Feature Sets

The current model relies on structured variables such as city, crime type, victim age, gender, weapon used, and year. However, crimes are complex social phenomena influenced by multiple latent factors. Integrating additional temporal features (e.g., time of day, month, public holidays), spatial context (e.g., GPS coordinates, neighbourhood population density), and socio-economic indicators (e.g., literacy rates, employment status, poverty levels) would provide a richer context and could substantially improve model performance. These additional attributes can be sourced from government census records, meteorological departments, and urban planning datasets.

B. Dynamic Temporal Modeling

Currently, the model treats temporal data statically. However, crime patterns often evolve with time and are best captured using time-series forecasting methods.

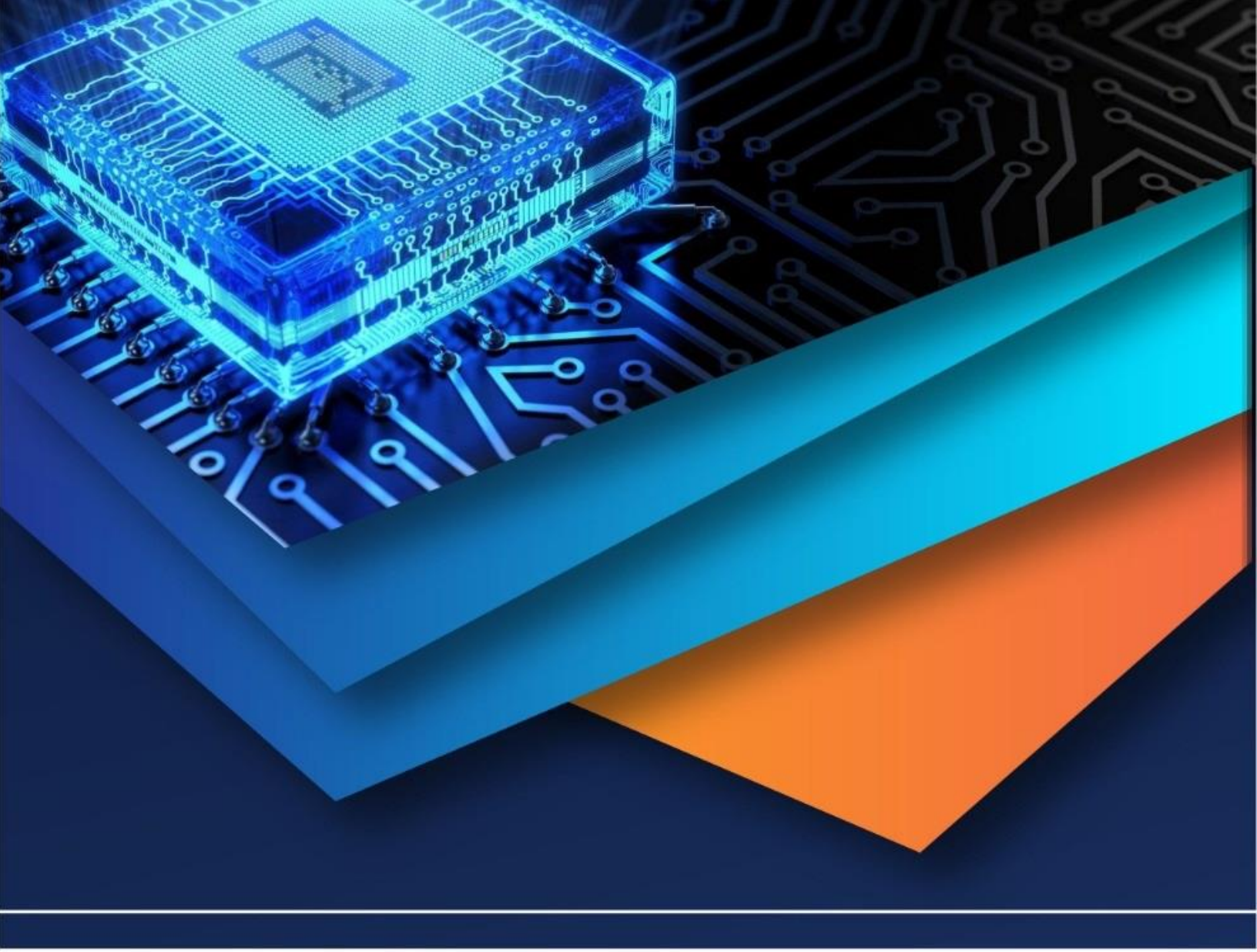
Future work may involve implementing algorithms such as ARIMA, Facebook Prophet, or LSTM (Long Short-Term Memory) networks to predict not just the crime domain but also the expected frequency and trend of crimes in a given region over time. These models will empower authorities to forecast high-risk months or years, enabling preventive actions in advance.



X. CONCLUSION

Crime continues to pose a complex and dynamic challenge to urban security and public policy. Traditional crime analytics based on manual records and tabular summaries are no longer sufficient to predict, prevent, or respond to criminal activity in real time. With the exponential growth of data collected from police departments, CCTV, social media, and public reporting systems, the need for automated, data- driven, and intelligent crime forecasting systems has become urgent. This study presents a significant contribution toward that goal by developing a practical, interpretable, and deployable crime domain prediction model tailored to the Indian context.

The main goal of this research was to build a supervised learning model using structured Indian crime data to classify incidents into categories like violent, property-related, or cybercrime. A Random Forest Classifier was selected for its robustness, interpretability, and ability to work with mixed data types. The model performed well, achieving 88% accuracy with balanced precision and recall, confirming its suitability for real-world crime prediction.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY