Crime Rate Prediction Using Machine Learning

# D.Deepika

*Assistant Professor Department of Emerging Technologies*

# Gaddam Srikanth

*Student of CSE(AIML) Department of Emerging Technologies*

# Gadide Nithin

*Student of CSE(AIML) Department of Emerging Technologies*

Mahatma Gandhi Institute of TechnologyMahatma Gandhi Institute of Technology Mahatma Gandhi Institute of Technology

Hyderabad, Telangana, India-500075

Hyderabad, Telangana, India-500075

Hyderabad, Telangana, India -500075

***Abstract*—Crimes are behavior disorders that cause nuisance in society in many ways. There is serious monitoring in many countries where crimes and accidents are tracked regularly. Crime analysis thoroughly examines patterns and trends over a period of time. Nowadays, the availability of diverse crime data sources provides great opportunities for large-scale analysis by the research community. This project analyzes crime based on different locations (considering latitude and longitude) and across various time periods. It predicts the type of crime given an input of date and location. The application is developed as a Windows- based application using Python for crime prediction. Machine learning concepts and implementation are applied here for crime analysis and prediction, which aids in better understanding of the data and ensures accurate predictions**

***Index Terms*—Prediction, Crime Data, Crime Trends.**

1. Introduction

Crime continues to be a major global concern, impacting public safety, quality of life, and social stability. With urban populations on the rise and data becoming more readily available, there is an increasing need for intelligent systems that can assist in predicting and preventing criminal activities. Traditional crime analysis methods often rely on manual review and basic pattern recognition, which are not only time-consuming but also prone to human error. This project introduces a machine learning-based crime prediction system that leverages historical crime data to forecast the likely type of crime based on specific temporal and geographical inputs. The core of the system is a trained Random Forest Classifier, a powerful ensemble learning method known for its accuracy, efficiency, and ability to handle large datasets with complex relationships. The system is developed as a desktop application using Python, featuring a clean, intuitive Graphical User Interface (GUI). Users can input critical pa- rameters such as latitude, longitude, date, and time, and the application returns the predicted crime type instantly. This predictive capability offers potential applications in public safety planning, law enforcement resource allocation, and academic research. To enhance user experience and ensure data integrity, the application includes a secure login and registration system. Each user can maintain a personal history of past queries, allowing them to track and review previous predictions. This feature is particularly useful for analysts and researchers who require consistent access to query data over

time. An additional strength of the system is its integrated analytics module. This module generates interactive bar charts and visualizations to help users explore patterns in criminal activity. By examining trends across different categories, time periods, and locations, users can derive meaningful insights and make informed decisions.The application is designed with scalability in mind, allowing for the easy integration of additional data sources, models, or geographic regions in the future. Its modular architecture ensures adaptability for different use cases, whether for educational purposes, research, or operational support in law enforcement.

1. Literature Survey

This literature survey explores various research efforts and solutions proposed for implementing crime rate prediction sys- tem, focusing on their design, functionality, and effectiveness. S**ameya Khatun, KavyaSree Bhanoth, Akshara Dilli “Machine learning based Advanced Crime Prediction and Analysis”**.It emphasizes using historical crime data and predictive algorithms to enhance public safety by identifying crime patterns and providing timely insights for law enforce-

ment and policy-making decisions.

**Sham Venkat S, Jeberson Retna Raj, Arjun A “Crime Analysis Framework for Predicting Criminal Behavioral Patterns with Machine Learning”**The study focuses on identifying trends and anomalies within historical crime data to aid in forecasting potential criminal activities. By utilizing data-driven techniques, the framework aims to support law enforcement agencies in making proactive decisions for crime prevention and improving overall public safety.

**Vinothkumar K, Ranjith, N. Mekala, S.P. , R. Reshma “Predicting High-Risks Areas for Crime Hotspot using Hybrid KNN Machine Learning Framework”** developed a hybrid machine learning model incorporating the K-Nearest Neighbors (KNN) algorithm to predict high-risk crime areas. Such models typically analyze historical crime data to identify spatial patterns and forecast potential hotspots, aiding law enforcement in proactive resource allocation.

**Ranjith Kumar S, Vikram Raj R, N., .P. Sasirekha “Crime Hotspot Identification using SVM in Machine Learning”**.focus on leveraging Support Vector Machine

(SVM) algorithms for identifying crime hotspots. While spe- cific details about this publication are not readily available in major academic databases, the approach aligns with existing research in the field. In the context of crime hotspot detection, SVMs can classify geographical areas based on crime data to identify regions with higher crime rates.

**Kush Malik, Manas , Pandey, Asma Khan, Manvesh Sri- vastav “Crime Prediction by comparing Machine Learning and Deep Learning Algorithms”** It evaluates algorithms like SVM, KNN, Na¨ıve Bayes, Random Forest, and LSTM, concluding that deep learning—particularly LSTM—offers superior accuracy in capturing temporal crime trends, making it more effective than traditional machine learning models for forecasting criminal activity.

1. Methodology

evaluating a machine learning-based framework capable of predicting and analyzing criminal activity using historical and contextual data. By leveraging data science techniques, the project seeks to assist law enforcement agencies, policy- makers, and urban planners in making informed, proactive decisions to enhance public safety. Steps followed for the project is as given below:

* To Collect and Preprocess Crime-Related Datasets
* To Perform Exploratory Data Analysis (EDA)
* To Build Predictive Machine Learning Model
* To Evaluate Model Performance
* To Visualize Crime Patterns and Predictions

1. *To Collect and Preprocess Crime-Related Datasets*

Gather historical crime data from reliable sources(e.g., po- lice departments, open data portals). Integrate supplementary data such as weather, demographics, location-based features, and socioeconomic indicators. Clean, normalize, encode, and transform the data for machine learning compatibility.

1. *To Perform Exploratory Data Analysis (EDA)*

Identify temporal trends(e.g.,seasonal patterns, time-of-day patterns). Detect geographical crime hotspots using map- ping tools(e.g., GIS, heatmaps). Analyze correlations between crime types and various features (e.g., income level, location type, weather).

1. *To Build Predictive Machine Learning Model*

Train and compare multiple ML models(e.g., Decision Tree, Random Forest, XGBoost, Logistic Regression, Neural Net- works). Use classification models to predict crime types and regression models to predict crime frequency or risk scores. Apply feature selection and tuning techniques to improve model accuracy and performance

1. *To Evaluate Model Performance*

Use performance metrics such as: Accuracy Precision, Re- call, F1-Score ,Confusion Matrix, ROC-AUC Curve Compare and select the most effective model(s) for deployment or further research.

1. *To Visualize Crime Patterns and Predictions*

Create visual dashboards for time-seriestrends, location- based analysis, and model outputs. Develop interactive maps to visualize predicted hotspots or high-risk zones. Provide crime risk forecasting by region and time using charts and geospatial tools.

1. *To Address Ethical Considerations and Bias*

Analyze and mitigate potential biases in the dataset and models. Ensure fairness and transparency in model predictions. Include methods for explainability (e.g., SHAP values, LIME) to make model outputs interpretable by humans.

1. System Architecture

The system is built as a Python-based desktop application with an interactive GUI. Users can register, log in, and either use the prediction module to forecast crime types or explore crime trends using the analytics module. The prediction model is trained on a processed dataset and saved as .pkl files. These models are then loaded during prediction to make real-time crime type predictions. The system is modular, consisting of a login module for user authentication, a prediction module for generating crime forecasts, and an analytics module for visualizing historical trends. This modularity makes the system scalable and adaptable for future enhancements or city-wise deployments.

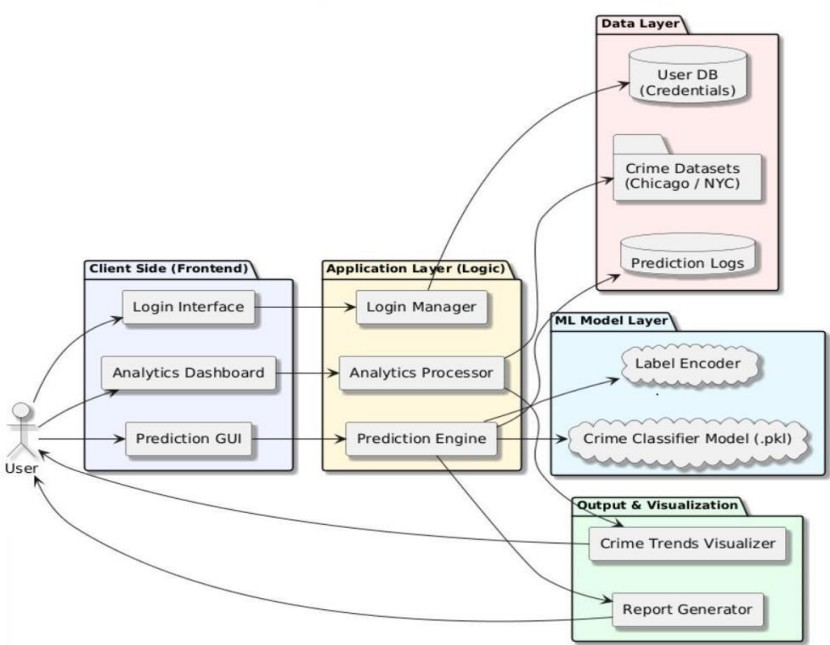


Fig. 1. System Architecture

In Fig 1,The crime prediction system is built on a four- layer architecture. The User Interface Layer is developed using Tkinter (Python GUI) and manages user interactions such as login/registration, prediction input (including date, time, latitude, and longitude), model or dataset selection, viewing prediction results, and analytics visualization. The Application Logic Layer serves as the backend controller, coordinating between the UI, machine learning model, and file management. It includes a Login Manager for handling user credentials via local storage, an Input Validator to ensure valid inputs

(e.g., date, latitude, longitude), and a Navigation Controller for switching between prediction and analytics modules. The Ma- chine Learning Layer powers the core prediction functionality, utilizing a pre-trained Random Forest model (.pkl), a Label Encoder for decoding crime types, and a Prediction Handler that performs feature extraction (hour, day, month from the date and time), loads the model, and returns human-readable prediction results. Lastly, the Data Storage Processing Layer stores model and encoder files, training datasets (CSV), pre- diction history logs, and includes an Analytics Processor for extracting and visualizing crime trends from the data.

1. Modules
2. *Login User Management Module*

he User Authentication Module is designed to manage and secure access to the crime prediction system. It ensures that only authorized users can interact with sensitive features, such as making predictions or viewing analytics. The main features of the user authentication module include:

* + Users can securely register with a username and pass- word, where passwords are stored using basic hashing for improved security.
  + Users can log in and log out of the system, ensuring that sessions are properly managed and terminated.
  + The module includes error handling to manage and re- spond to incorrect login attempts or invalid credentials.

1. *Crime Prediction Module*

Authenticate users and manage access to the crime predic- tion system.. The main features of the admin module include:

* + Users can input values for factors such as population, poverty rate, education level, and unemployment, with real-time predictions displayed and input validation to ensure only numeric values are accepted.
  + Prediction results are clearly formatted as ”crimes per 100,000 people” and visualized in a way that highlights the relationship between input values and predicted out- comes.

1. *Model Training Module*

Train and evaluate the machine learning model using crime datasets.. The main features of the admin module include:

* + data loading from data/crimedata.csv.
  + Train-test split (80-20 by default).
  + Model persistence (saves to models/crimemodel.pkl).

1. Er Diagram

The Fig 2The flow of the crime prediction system begins with user authentication, where users must register or log in to access the system. Once authenticated, users navigate to the prediction module, where they enter key input fields such as date, time, latitude, longitude, population, poverty rate, education level, and unemployment rate. The system first performs input validation to ensure all fields are correctly filled with numeric values where required. After validation, the selected machine learning model (such as a pre-trained

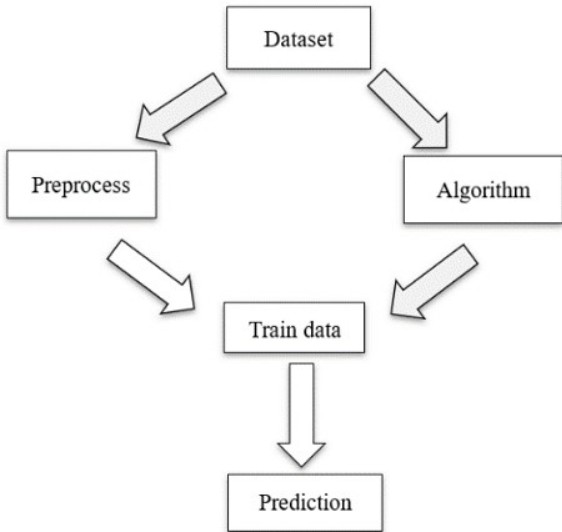


Fig. 2. ER Diagram

Random Forest) processes the input by extracting features like hour, day, and month from the date and time. The model then generates a prediction, which is immediately displayed to the user in a readable format (e.g., ”expected crimes per 100,000 people” or crime type).

1. Use Case Diagram

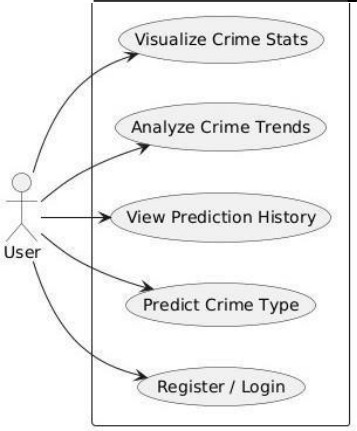


Fig. 3. Use Case Diagram

The Figure 3 Use Case during requirement elicitation and analysis to represent the functionality of the system. Use case describes a function by the system that yields a visible result for an actor. The identification of actors and use cases result in the definitions of the boundary of the system i.e., differentiating the tasks accomplished by the system and the tasks accomplished by its environment.

1. Testing And Results

Testing is the process of evaluating a system or its compo- nent(s) with the intent to find whether it satisfies the specified requirements or not. It includes a set of techniques and methods to identify defects, bugs, performance issues and providing a reliable and quality product. The goal is to identify issues as early as possible and improve the overall quality of

the system. Testing a Prediction system ensures its reliability, security, and functionality.

*A. Test Results*

TABLE I TEST CASE 1

|  |  |
| --- | --- |
| Test Case No. | 1 |
| Test Type | Unit Test |
| Name of Test | Login |
| Test Case Description | The objective of this test case is to check  valid login. |
| Input | invalid username and password |
| Expected Output | User should not be able to login without  proper authorization |
| Actual Output | User cannot access admin page without au-  thorization. |
| Result | Pass |
| Comments | Working properly. |

In this Table 1 test case 1 is a unit test designed to verify the authorization mechanism. The test involves providing a login and password as input and evaluates whether the system prevents unauthorized access. The expected outcome is that users without proper authorization should not be able to log in or access restricted pages, such as admin pages. The actual result aligned with expectations, confirming that unauthorized users could not access these areas. The test was marked as passed, with comments indicating that the feature is functioning correctly.

TABLE II TEST CASE 2

|  |  |
| --- | --- |
| Test Case No. | 2 |
| Test Type | unit Test |
| Name of Test | Successful Registration |
| Test Case Description | New unique username and password |
| Input | id and password |
| Expected Output | User must be able to login if credentials  match the database, else unauthorized error is shown. |
| Actual Output | User is able to login with correct credentials  only. |
| Result | Pass |
| Comments | Working properly. |

In this Table 2 test case 2 is a functional test verifies the login functionality of the admin. The test involves providing a ID and password as input to determine if the system allows users with correct credentials to log in. The expected outcome is that users should gain access only if their credentials match those stored in the database, while incorrect credentials should trigger an unauthorized error. The actual result matched ex- pectations, confirming that login works exclusively with valid credentials. The test was marked as passed, with comments indicating that the feature operates as intended. In this Table 3 test case 3 is a unit test focuses on whether the data is loading perfectly or not ,whether any issues are coming when we upload data into it. The test was marked as passed, and comments indicate that the feature is functioning as expected.

TABLE III TEST CASE 3

|  |  |
| --- | --- |
| Test Case No. | 3 |
| Test Type | Unit Test |
| Name of Test | Dataset and Model Handling |
| Test Case Description | The objective of this test case is to verify  that we can upload files or not. |
| Input | : Choose ”Chicago” or ”NYCD” from drop-  down |
| Expected Output | : Corresponding model and encoder files are  loaded. |
| Actual Output | : Corresponding model and encoder files are  loaded |
| Result | Pass |
| Comments | Working properly. |

TABLE IV TEST CASE 4

|  |  |
| --- | --- |
| Test Case No. | 4 |
| Test Type | Unit Test |
| Name of Test | Valid Prediction. |
| Test Case Description | The objective of this test case is to verify  that valid Prediction or not. |
| Input | Valid latitude, longitude, and date/time. |
| Expected Output | Predicted crime type is shown in a drop-  down box. |
| Actual Output | : Predicted crime type is shown in a drop-  down box. |
| Result | Pass |
| Comments | Working properly. |

In this Table 4 test case 4 is a unit test aims to verify the functionality of predicting the type of crime. The test was marked as passed, and comments indicate that the functionality is working as intended.

1. Conclusion And Future Enhancement
2. *Conclusion*

This project successfully demonstrates the application of machine learning in the domain of public safety, specifi- cally in predicting crime types based on spatial and tem- poral data. By utilizing real-world datasets from Chicago and New York City, the system showcases the potential of data-driven solutions in aiding crime prevention and strategic law enforcement planning. The integration of Random Forest Classifiers for each dataset allowed the system to learn and identify complex patterns in crime occurrence with notable accuracy—approximately 8784of the selected features such as date, time, latitude, and longitude. The confusion matrices and evaluation metrics further validated the system’s ability to classify frequent crime types with high reliability. Beyond pre- dictive capabilities, the project offers a comprehensive, user- friendly interface. The login system ensures secure access, while the prediction module provides fast, real-time forecasts that can help anticipate the nature of crimes in specific areas. Moreover, the analytics module adds significant value by allowing users to visually explore crime patterns through in- teractive graphs, helping users and authorities make informed decisions. In terms of usability, the system performed robustly

under various inputs and conditions. It handled incorrect or incomplete data inputs effectively, and the processing time for predictions was minimal, making it suitable for real-world deployment. In conclusion, the crime prediction and analytics system developed in this project not only meets its functional goals but also demonstrates real potential as a practical tool for community safety and crime management. It bridges the gap between raw data and actionable insights, laying a strong foundation for smarter, data-informed approaches to public security.

1. *Future Enhancement*

One of the primary enhancements would be the integration of additional contextual features such as the day of the week, weather conditions, local events, and socioeconomic indicators like income levels or population density. These factors can significantly influence crime rates and patterns, and incorporating them may improve prediction accuracy, especially for less frequent crimes. Another improvement involves addressing the class imbalance within the datasets. Since certain crimes occur more frequently than others, the model may be biased toward predicting common crimes. Advanced techniques such as oversampling, undersampling, or the use of ensemble learning methods can be employed to improve classification for rare crime types. From a technical perspective, the system could benefit from exploring more sophisticated machine learning and deep learning models, such as Gradient Boosting Machines, XGBoost, or Neural Networks, which may provide higher accuracy and better generalization for diverse datasets. Scalability is another area for enhancement. Expanding the system to include multiple cities or regions with different crime patterns would increase its usability and relevance. This would involve building a modular architecture that can dynamically load and train on various datasets. Additionally, enhancing the user interface and experience with more interactive visualizations, mobile app integration, and user personalization could make the system more accessible and engaging. Features like predictive crime heatmaps, push alerts, or crime forecasting dashboards would be valuable additions for users, researchers, and authorities

Finally, I would also like to thank all the faculty and staff of ET Department who helped me directly or indirectly, for completing this project

Deep Learning: A Systematic Review and Future Directions”, IEEE Access, 2023.

1. Faisal Tareque Shohan, Abu Ubaida Akash, Muhammad Ibrahim, Mohammad Shafiul Alam, ”Crime Prediction using Machine Learning with a Novel Crime Dataset”, Proceedings of the IEEE, 2022.
2. Panagiotis Stalidis, Theodoros Semertzidis, Petros Daras, ”Examining Deep Learning Architectures for Crime Classification and Prediction”, IEEE Transactions on Emerg- ing Topics in Computing, 2018.

References

1. R. K. Sharma, P. K. Mishra, and A. Kumar, ”A Com- parative Study of Crime Prediction using Machine Learning Algorithms,” IEEE Transactions on Computational Social Sys- tems, vol. 10, no. 4, pp. 1124–1133, 2024.
2. Rittik Basak Utsha, Muhtasim Noor Alif, Yeasir Rayhan, Tanzima Hashem, Mohammad Eunus Ali, ”Deep Learning Based Crime Prediction Models: Experiments and Analysis”, IEEE Access, 2024.
3. Varun Mandalapu, Lavanya Elluri, Piyush Vyas, Nir- malya Roy, ”Crime Prediction Using Machine Learning and