World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(RESEARCH ARTICLE)

Check for updates

Crime data analysis and prediction of arrest using machine learning

Revanth Sankul, Tejaswi Reddy Aruva *, Sai Varun Kankal, Greeshma Arrapogula and Shoeib Khan Mohammed

Department of CSE (Data Science), ACE Engineering College, Hyderabad, Telangana, India.

World Journal of Advanced Research and Reviews, 2025, 25(02), 498-506

Publication history: Received on 26 December 2024; revised on 01 February 2025; accepted on 04 February 2025

Article DOI: https://doi.org/10.30574/wjarr.2025.25.2.0385

Abstract

In order to forecast future criminal activity and improve law enforcement tactics, crime data analysis and arrest prediction entails looking at past crime data to find patterns and trends. This area analyzes a variety of variables, including time, place, demography, and the kinds of crimes committed, using statistical methods, machine learning algorithms, and data mining. The objective is to give law enforcement organizations useful information so they may better allocate resources, determine crime, and enhance public safety. It entails combining data from multiple sources, such as arrest logs, crime reports, socioeconomic information, and even environmental elements like urbanization and weather trends. To comprehend how crime trends change over time, sophisticated analytical methods such as random forest is used in predicting the arrests.

Keywords: Machine Learning; Predicting the arrest using Random forest Algorithm; User friendly stream lit interface; Statistical methods; Crime Reports; Crime data Analysis

1. Introduction

Crime is a serious social problem that has an huge impact on people's safety and wellbeing both individually and collectively. It is now simpler to identify trends and forecast criminal activity because of technological advancements and the wealth of crime data.

It might be difficult to extract useful insights from massive amounts of complex data when using traditional crime analysis techniques. By identifying hidden patterns in crime data and forecasting significant consequences, including the probability of arrests, machine learning (ML) provides a potent remedy. ML models can help law enforcement agencies make proactive decisions by identifying correlations between variables such as crime type, time, location, and arrest rates by utilizing historical crime records. Developing a machine learning-based system to analyze crime data and forecast arrests is the goal of our study.

2. Related Work

Decision Trees were preferred for their interpretability, while SVM showed high accuracy in complex datasets with non-linear relationships.

The application of classification techniques, including Decision Trees and Support Vector Machines (SVM), for crime prediction was the subject of one of the first studies. These models were used to identify and categorize different kinds of crimes and determine the probability of an arrest.

* Corresponding author: , A Tejaswi

A lot of research has been done recently on the use of machine learning in the analysis of crime data and the prediction of arrests. A lot of research work have shown how machine learning models can be used to identifying crimes.

Recent research has said to include models such as Random Forest and Gradient Boosting Machines (GBM), which have proven to be more robust and accurate in crime prediction tasks. These models handle data imbalances better and provide improved prediction capabilities by aggregating multiple decision trees.

Methods like Spatial crime pattern analysis has made extensive use of clustering methods, especially K-Means Clustering Algorithm and DBSCAN. These methods have been crucial in locating crime hotspots and assisting law enforcement in more efficiently allocating resources.

Studies integrating Geographic Information Systems (GIS) with machine learning models have further enhanced crime prediction by offering geospatial insights

Another key area of research has focused on feature engineering and selection Research has shown that a number of variables, including the type of crime, the time of the incident, the use of a weapon, and the location, have a substantial impact on the results of arrests. It has been demonstrated that efficient feature selection methods increase the precision and effectiveness of machine learning models.

Despite these advancements, several challenges remain. Data privacy and security concerns limit the availability of comprehensive crime datasets. Additionally, biases in historical crime data can lead to discriminatory outcomes in machine learning models. Researchers have proposed methods such as bias correction techniques and synthetic data generation to mitigate these issues.

This project is build upon the using existing body of work by integrating advanced machine learning techniques to predict arrest outcomes more accurately. By using a combination of classification models and feature engineering methods, the study aims to provide actionable insights for law enforcement agencies, contributing to more efficient crime prevention strategies and safer communities.

3. Existing System

The existing system for crime data analysis typically involves the use of traditional statistical methods and historical crime data to identify trends and patterns. Law enforcement agencies often rely on manual data analysis and simple query-based reports to understand crime occurrences and allocate resources.

This approach, while useful, has limitations in its ability to predict future crime incidents with high accuracy and often lacks real-time insights. The reliance on historical data alone can result in reactive rather than proactive measures, potentially missing emerging crime trends.

4. Proposed Model

The proposed system advances crime data analysis by integrating machine learning techniques and advanced predictive modeling to forecast arrest probabilities based on various crime attributes. This proactive and data-driven approach promises to improve public safety and operational efficiency in crime management.

By the using the advanced algorithms like Random Forest Classifier, the system improves prediction efficiency and accuracy, allowing law enforcement agencies to take a proactive stance in preventing crime. Strategic decision-making and efficient resource allocations are supported by more timely and nuanced insights made possible by the integration of many data sources, real-time updates, and sophisticated visualization tools.

5. Methodology

The methodology for criminal data analysis and arrest prediction involves data collection, preprocessing, feature selection, and model implementation using machine learning techniques. The system aims to analyze criminal patterns, predict the likelihood of arrests, and support law enforcement agencies with data-driven insights.

The machine learning algorithms and techniques selected for this project include libraries such as sklearn, pandas and numpy, matplotlib. Those Libraries imported are:

Figure 1 Libraries imported

5.1. Data Collection

Sources: Gather information from police department files, open data portals (like data.gov and city open data portals), and public crime databases (like the FBI's UCR and NIBRS). o

factors: Verify that the dataset contains pertinent factors including the type of crime, the location, the time it occurred, demographic data, and arrest specifics.

5.1.1. Data Integration

Consolidate information from several sources into a single dataset while eliminating duplicates and guaranteeing uniform formatting.

External Sources of Information: To capture a wider context and potential influences on crime rates, include data from other sources, such as social media, economic indicators, and weather conditions.

5.2. Dataset Selection:

Selecting the right dataset is crucial for training an effective machine learning model for criminal data analysis and arrest prediction. Below are some potential datasets and their key features:

5.2.1. Publicly Available Crime Datasets

Several datasets provide detailed crime reports, arrest records, and related information. Some well-known sources include:

- FBI Uniform Crime Reporting (UCR)
  - Covers violent and property crimes
  - Includes arrest data, offender demographics, and crime locations
  - Annual and monthly crime reports
- National Crime Victimization Survey (NCVS)
  - Includes crime reports from victims, even if no arrests were made
  - Details about crime circumstances, locations, and suspect information
- Chicago Crime Dataset (Kaggle, City of Chicago Open Data)
  - Records of reported crimes from 2001 to the present
  - Includes crime type, location, time, and arrest status
- New York Police Department (NYPD) Complaint Data
  - Comprehensive crime reports including felonies, misdemeanors, and violations
  - Information on crime type, suspect demographics, and location
- Los Angeles Crime Dataset (LAPD Open Data)
  - Provides crime type, date, location, and arrest details
  - Can be used for geospatial crime analysis

5.3. Data Preprocessing

Since crime data may contain missing values, inconsistencies, and noise, preprocessing is crucial. The steps include:

- Handling missing values using imputation techniques
- Removing duplicate and irrelevant records
- Normalizing and standardizing numerical features
- Converting categorical data (e.g., crime type, suspect gender) into numerical format using encoding techniques (e.g., One-Hot Encoding, Label Encoding)
- Text preprocessing for crime descriptions (tokenization, stemming, removing stopwords)

5.4. System Architecture

A robust system architecture for crime data analysis and prediction of arrests integrates multiple layers and components to ensure seamless data flow and advanced analytical capabilities. The design starts with data ingestion, where a variety of data sources are gathered via ETL procedures and API connections, including police records, public crime reports, demographic information, environmental elements, and social media feeds. Then, using tools like Apache Hadoop and Amazon Redshift, this data is kept in centralized data lakes and warehouses. Data cleaning, normalization, and feature engineering are examples of preprocessing procedures that get the data ready for analysis. Descriptive analytics and visualization are made possible by business intelligence (BI) technologies like Tableau and Power BI, which offer insights into past crime trends.

For predictive modeling, machine learning algorithms such as logistic regression, decision trees, and neural networks are employed. These models are trained and validated using historical data to predict the likelihood of arrests. Real-time data processing capabilities are achieved through stream processing tools like Apache Kafka and Apache Flink, enabling the system to update predictions and trigger alerts for high-risk situations promptly. Security measures, including data encryption and access control, ensure compliance with legal standards. The architecture is complemented by continuous monitoring and feedback loops to maintain and enhance model accuracy, while user interfaces, such as interactive dashboards and mobile applications, provide law enforcement agencies with actionable insights and predictive analytics for proactive crime management.

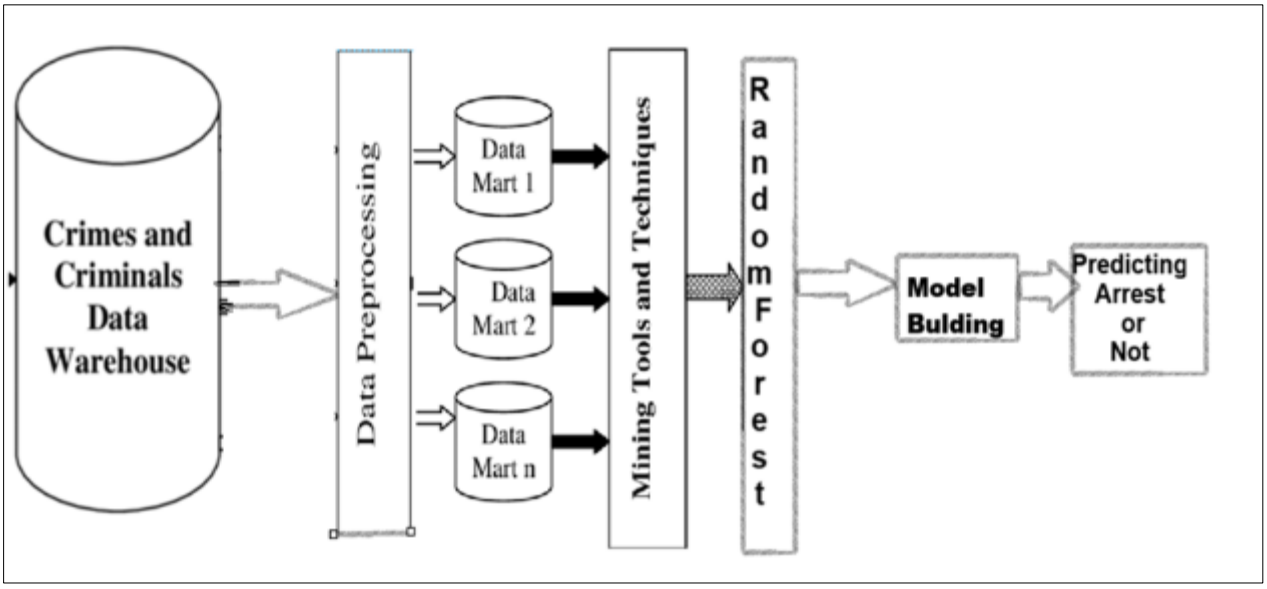

Figure 2 System Architecture

It illustrates how input data flows through various processing layers, resulting in meaningful outputs.

5.4.1. System Architecture Components

- Input Module

Collects crime data from **databases (FBI UCR, NCVS), real-time reports, GIS maps, and social media** for continuous updates.

- Data Processing & Feature Extraction

Cleans, preprocesses, and standardizes data.

Handles missing values, encodes categorical data, and selects key crime-related features.

- Crime Pattern Recognition & Arrest Prediction

analyzes crime trends and forecasts arrests using machine learning models (ML) such as logistic regression, random forests, xg boost, and neural networks**.**

- Confidence Scoring

Assigns reliability scores using probability metrics like Softmax, AUC-ROC.

- User Interface & Visualization

Displays crime trends, geospatial maps, arrest likelihood indicators, and historical analytics in a user-friendly dashboard.

- Output Module

Provides real-time alerts, arrest predictions, crime heatmaps, and supports proactive policing.

5.5. Model Development

The core of the system is the model, which is essential for examining crime trends and forecasting the possibility of arrests. To provide precise and trustworthy forecasts, it is created with cutting-edge machine learning (ML) algorithms

5.5.1. ***Validation****:*

To ensure the model performs well and makes meaningful predictions, the following validation steps are carried out:

- **Model Evaluation and Validation**:
  - **Metrics**: We use metrics such as accuracy, precision, recall, F1 score, ROC-AUC, and confusion matrix to assess model performance and get accuracy and predict the tasks.
  - **Cross-Validation**: Perform k-fold cross-validation to ensure model robustness and generalizability.
  - **Error Analysis**: Misclassifications are carefully analyzed to identify weaknesses in the model, enabling improvements and refinements for better results.

- **Training**:

Train the model using random forest classifier.

  - Random Forest can be highly effective in analyzing crime data and predicting arrests due to its ability to handle large datasets with many variables, deal with missing data, and avoid overfitting.
  - It works by constructing a large number of the individual decision trees during training and then combining their results to make predictions.

```
# Ensure the target variable 'Arrest' is binary (0 or 1)
crime_data['Arrest'] = crime_data['Arrest'].astype(int)

# Split the data into features and target variable (arrest)
X = crime_data[features]
y = crime_data['Arrest']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train a machine learning model (Random Forest Classifier, for example)
rf_classifier = RandomForestClassifier(n_estimators=100, random_state=42)
rf_classifier.fit(X_train, y_train)

# Make predictions on the test set
y_pred = rf_classifier.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)

# Generate classification report and confusion matrix
print(classification_report(y_test, y_pred))
print(confusion_matrix(y_test, y_pred))

# Visualize the confusion matrix
plt.figure(figsize=(8, 6))
cm = confusion_matrix(y_test, y_pred)
plt.imshow(cm, interpolation='nearest', cmap=plt.cm.Blues)
plt.title('Confusion matrix')
plt.colorbar()
plt.xticks([0, 1], ['No Arrest', 'Arrest'])
plt.yticks([0, 1], ['No Arrest', 'Arrest'])
plt.xlabel('Predicted')
plt.ylabel('True')
plt.show()
```

Figure 3 Model Training

- **Testing**

System testing comprises a series of diverse tests aimed at thoroughly exercising the computer-based system. While each test serves a distinct purpose, all are geared towards confirming that all system elements have been effectively integrated and are performing their designated functions. The testing process is conducted to ensurethat the product precisely fulfills its intended purpose. During the testing phase, the following objectives are pursued:

- To affirm the quality of the project.
- To identify and rectify any lingering errors from previous stages.
- To validate the software as a solution to the initial problem.
- To ensure the operational reliability of the system.

- **Testing Methodologies**

There are many different types of testing methods or techniques used as part of thesoftware testing methodology. Some of the important testing methodologies are:

- Unit testing
- System testing

5.6. Real-Time Data Processing

5.6.1. Collecting data in real time

Data Sources: CCTV cameras, police reports, GPS devices and social media updates.

This provides the continuous information for analysis.

5.6.2. Cleaning and Preparing Data

Removing Errors: Unnecessary or wrong information is filtered out.

Standardizing data: Information such as crime types, locations is in organized format.

5.6.3. Machine learning models for prediction:

Classification models: Algorithms like Random Forest classifier is used.

5.6.4. Handling Live Data

Tools like Apache Kafka and Apache Spark process live data efficiently.

5.6.5. Real-Time Alerts & Notifications

- Heatmaps & Dashboards: Crime hotspots & suspect movement tracking.
- Law Enforcement Alerts: Automated warnings for high-risk areas & offenders.

This ML-driven system enhances crime prevention, real-time surveillance, and proactive policing.

5.7. Visualization

Delivering results in a user-friendly manner is crucial for system usability.

5.7.1. Real-Time Display:

- GUI Development for Real-Time Display

Features:

- Tool Used: Stream lit, Flask, or Dash for interactive visualization.
- Heatmaps: Highlight high crime area based on previous and recently available data.
- Interactive Filters: Allows users to explore data by location, time, crime type.
- Data Visualization: The data like crime patterns, arrest types, arrest rate, statistical trends displayed in charts and heatmap format

- Crime Pattern Detection Using ML
  - Crime Type Prediction: Classification algorithm like random forest classifier is used to predict the likely outcomes.
  - Arrest Prediction Models: Logistic Regression is used to estimate the arrest rate probability
- ML-Based Crime & Arrest Prediction
  - Crime Pattern Analysis:
    - ✓ K-Means Clustering for crime hotspot detection.
    - ✓ Decision Trees, XG Boost for identifying high-risk suspects.
  - Arrest Likelihood Forecasting:
    - ✓ LSTM, ANN analyze behaviour, past records, and crime frequency.
    - ✓ Logistic Regression predicts probability of arrest.
- Real-Time Alerts & Law Enforcement Dashboard
  - Risk Scores & Alerts: Notifies officers of suspicious activities.
  - Crime Trend Dashboards: Shows dynamic analytics for better decision-making.

6. Results and Discussion

Figure 4 User Interface

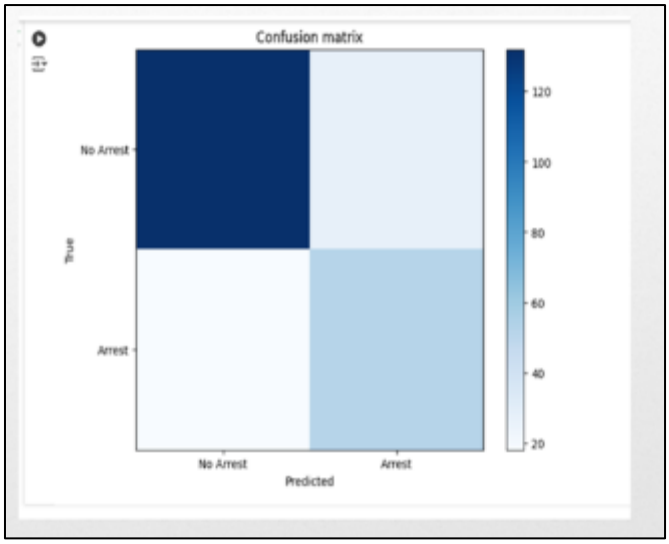

Figure 5 Result image of confusion matrix

7. Conclusion

The experiment on analyzing crime data and predicting arrests shows how data science and machine learning can be powerfully applied to improve law enforcement and public safety. The project successfully created a predictive model that can estimate the likelihood of arrests based on different criminal variables by utilizing historical crime data and techniques like data pretreatment, feature engineering, and machine learning modeling. Accuracy, classification reports, and confusion matrices were among the strict evaluation measures used in conjunction with the Random Forest Classifier to guarantee the model's strong and dependable performance. Matplotlib and other visualization tools were used to give a clear picture of the model's performance. In addition to demonstrating predictive analytics' potential for preventing crime, this initiative emphasizes the significance.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] S. Kim, P. Joshi, P. S. Kalsi, and P. Taheri, "Crime Analysis Through Machine Learning," in 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 1-3 Nov. 2018 2018, pp. 415-420, doi: 10.1109/IEMCON.2018.8614828.

[2] Y. Lin, T. Chen, and L. Yu, "Using Machine Learning to Assist Crime Prevention," in 2017 6th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI), 9-13 July 2017 2017, pp. 1029-1030, doi: 10.1109/IIAI-AAI.2017.46.

[3] Pratibha, A. Gahalot, Uprant, S. Dhiman, and L Chouhan, "Crime Prediction and Analysis," in 2nd International Conference on Data, Engineering and Applications (IDEA), 28-29 Feb. 2020 2020, pp. 1-6, doi: 10.1109/IDEA49133.2020.9170731.

[4] A. Jawla, M. Singh, and N. Hooda, "Crime Forecasting using Folium," Test Engineering and Management, vol. 82, pp. 16235-16240, 05/31 2020.

[5] W. Gorr and R. Harries, "Introduction to crime forecasting," International Journal of Forecasting, vol. 19, no. 4, pp. 551-555, 2003/10/01/ 2003, doi:https://doi.org/10.1016/S0169-2070(03)00089-X.

[6] M. Granroth-Wilding and S. Clark, "What Happens Next? Event Prediction Using a Compositional Neural Network Model," in AAAI, 2016.

[7] E. Ahishakiye, E. Opiyo, and I. Niyonzima, "Crime Prediction Using Decision Tree (J48) Classification Algorithm," International Journal of Computer and Information Technology (ISSN: 2279 – 0764), 05/15 2017.

[8] H. Nguyen, C. Cai, and F. Chen, "Automatic classification of traffic incident's severity using machine learning approaches," IET Intelligent Transport Systems, vol. 11, 07/14 2017, doi:10.1049/iet-its.2017.0051.

Author's short biography

Mr Revanth Sankul: I am Revanth Sankul, an assistant professor at ACE Engineering College, specializing in Computer Science and Engineering (Data Science). I am passionate about guiding students and fostering innovation in emerging technologies. I am dedicated to continuously enhancing my teaching methodologies and contributing to research in my field.	
Tejaswi Reddy Aruva: I am Tejaswi Reddy Aruva, a final-year B.Tech student at ACE Engineering College, specializing in Computer Science and Engineering (Data Science). I am passionate about data science and programming, and I enjoy discovering emerging technologies to expand my expertise. I am committed to improving my skills and applying them to solve real-world challenges in my field	
Sai Varun Kankal: I am Sai Varun Kankal, a final-year B.Tech student at ACE Engineering College, specializing in Computer Science and Engineering (Data Science). I have a strong interest in programming and emerging technologies, and I aim to continuously improve my knowledge to address practical challenges in the tech world.	
Greeshma Arrapogula: I am Greeshma Arrapogula, a final-year B.Tech student at ACE Engineering College, specializing in Computer Science and Engineering (Data Science). I am enthusiastic about data science and programming, and I thrive on discovering new technologies. I am dedicated to refining my skills and leveraging them to address complex problems	
Shoeib Khan Mohammed I am Shoeib Khan Mohammed, a final-year B.Tech student at ACE Engineering College, specializing in Computer Science and Engineering (Data Science). My passion lies in exploring data-driven solutions and staying updated with emerging technologies. I am focused on honing my technical skills and solving practical challenges