

# ML\_RESEARCH\_PAPER\_Final.docx

 Vishwakarma Group of Institutions

---

## Document Details

### Submission ID

trn:oid:::3618:91613784

### Submission Date

Apr 17, 2025, 9:05 AM GMT+5:30

### Download Date

Apr 17, 2025, 9:07 AM GMT+5:30

### File Name

ML\_RESEARCH\_PAPER\_Final.docx

### File Size

5.0 MB

12 Pages

7,552 Words

45,150 Characters





# 3% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.




## Filtered from the Report

- Bibliography
- Quoted Text
- Cited Text
- Small Matches (less than 9 words)

## Match Groups


-  **22 Not Cited or Quoted 3%**  
Matches with neither in-text citation nor quotation marks
-  **0 Missing Quotations 0%**  
Matches that are still very similar to source material
-  **0 Missing Citation 0%**  
Matches that have quotation marks, but no in-text citation
-  **0 Cited and Quoted 0%**  
Matches with in-text citation present, but no quotation marks

## Top Sources

- 3%  Internet sources
- 2%  Publications
- 2%  Submitted works (Student Papers)

## Integrity Flags





### 1 Integrity Flag for Review

-  **Replaced Characters**  
10 suspect characters on 2 pages  
Letters are swapped with similar characters from another alphabet.




Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

## Match Groups

-  **22 Not Cited or Quoted 3%**  
Matches with neither in-text citation nor quotation marks
-  **0 Missing Quotations 0%**  
Matches that are still very similar to source material
-  **0 Missing Citation 0%**  
Matches that have quotation marks, but no in-text citation
-  **0 Cited and Quoted 0%**  
Matches with in-text citation present, but no quotation marks

## Top Sources

- 3%  Internet sources
- 2%  Publications
- 2%  Submitted works (Student Papers)

## Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

- 1** Internet  
**www.mdpi.com** <1%
- 2** Submitted works  
**Letterkenny Institute of Technology on 2024-01-04** <1%
- 3** Publication  
**Yelshetty, Neha. "FairTrace : An Automated Unsupervised Fairness Machine Lear..."** <1%
- 4** Submitted works  
**Southern Arkansas University (Blackboard LTI 1.3) on 2024-10-31** <1%
- 5** Internet  
**scipublication.com** <1%
- 6** Internet  
**www.bpasjournals.com** <1%
- 7** Internet  
**docplayer.net** <1%
- 8** Internet  
**glisc.info** <1%
- 9** Internet  
**www.statista.com** <1%
- 10** Publication  
**Glenn Grana, James Windell. "Crime and Intelligence Analysis - An Integrated Rea..."** <1%

## 11 Submitted works

University of Hertfordshire on 2024-08-18 <1%

---

## 12 Internet

hdl.handle.net <1%

---

## 13 Internet

technodocbox.com <1%

---

## 14 Publication

R. N. V. Jagan Mohan, B. H. V. S. Rama Krishnam Raju, V. Chandra Sekhar, T. V. K. P... <1%

---

## 15 Submitted works

University of Hull on 2025-03-06 <1%

---

## 16 Submitted works

University of Northumbria at Newcastle on 2023-09-21 <1%

---

## 17 Internet

www.theebi.org <1%

---

## 18 Publication

Álvaro Briz-Redón, Jorge Mateu, Francisco Montes. "Modeling the Influence of Pla... <1%

# Crime Severity Prediction from Real Crime Data with Machine Learning

Shrey Arun Ingole, Shailesh Yerawad, Soham Godkar, Pranav Waklekar

Dr. Ratnamala Bhimanpallewar, Dr. Pravin Futane

<sup>1-6</sup>*Vishwakarma Institute of Information Technology, Pune, India.*

**Abstract:** Crime analysis is an essential discipline in contemporary policing, allowing police agencies to analyze patterns of criminal behavior through a variety of data-research approaches. The current project proposed a framework for the integrated analysis of crime data by linking Indian Penal Code (IPC) and Non-IPC sections of law to descriptive crime types, while assessing the relative severity of crimes through machine-learning methodologies. A crime records dataset containing features of date, time, location and legal sections was extracted from multiple police stations. The legal sections were mapped to specific crimes through previously established codebooks and regular expressions, and were pre-processed to create temporal variables (e.g., hour and day) and spatial variables (e.g., distance) to police stations. Crime severity was computed based on crime type, time of crime, and proximity to police, and grouped into levels of severity (e.g., "Low" through "Very High"). A Random Forest Classifier was used to achieve an overall prediction accuracy of 92%. Findings indicated that distance from a police station and certain types of crime are the most consistent predictors of crime severity. This framework offers a novel way to analyze patterns of crime and predicting crime severity, and can be utilized by police services to assist in resource allocation and public safety in a scalable framework. Future research may include the expansion of the dataset and exploring alternative predictive modeling.

**Keywords:** Random Forest Classifier, IPC Mapping, Crime Severity Prediction, Data Preprocessing, Law Enforcement

## 1. INTRODUCTION

For a long time, crime analysis has been an important tool for law enforcement relying on manual data collection processes, and basic statistical techniques to reveal patterns and trends. For example, mapping crime hotspots using Geographic Information Systems (GIS), or monitoring crime rates over time, have been the standard approaches (Chainey and Ratcliffe, 2005). These approaches have been somewhat successful; however, they tend to be resource intensive, their applicability in real time is limited, and they do not effectively address nuanced changes in crime behavior, especially in more complex legal environments such as India. In India, the Indian Penal Code (IPC) and Non-IPC acts, for example, provide legal differentiators of crime behavior, that traditional methods requiring manual appraisal, are completely unable to tackle effectively, which limits the scalability of these approaches, and any derived inference.

In our research, we present a new, automated process to address these limitations by converting IPC (Indian Penal Code) and Non-IPC legal clauses into descriptive categories of crime via a predefined dictionary and regular expressions. By employing reference files analogous to `ipc_mapping` and `non_ipc_mapping`, we take the raw legal codes and distil them into categories of crime that can be analyzed. We process the comprehensive, crime data set—including variables related to Date of Occurrence, Time of Occurrence, Longitude, Latitude, and Station—to extract important temporal (e.g., hour, day of week) and spatial (e.g., distance from station)

features. A unique aspect of our method is estimating crime severity as a product of crime type and contextual frameworks, such as time of day and distance to law enforcement, based on values contained in crime weights. Features generated, which were predicted using a Random Forest Classifier model, provided a 92% accuracy predicting crime severity (as validated in our modeling script). This automated process not only simplifies the process of interpreting legal codes, but serves as a robust data-driven tool to thinking about crime.

Our method provides pathways for enhancements. For example, embedding real-time data feeds may allow for dynamic crime tracking; additionally, expanding the data to additional regions would broaden the scope of the method. More advanced algorithms, such as XGBoost, could also be explored to improve prediction accuracy. Overall, with greater scalability, our method provides law enforcement agencies the ability to better allocate resources and avoid crime before it occurs. This work represents progress within the field of crime analysis integrating automation and predictive modeling to assist in decision making and contribute to communities being safer.

## 2. RELATED WORK

For many years, traditional statistical methods and geospatial analyses have played a large role in crime analysis. Descriptive statistics (e.g., crime frequencies, peak crime times, and crime seasonality) is one of the earliest techniques developed to understand crime patterns [1]. Using correlation matrices, crime rates, and the effect of variables such as poverty or education

levels was explored. GIS-based crime mapping has been another foundational part of crime analysis. Mapping crimes in GIS allowed researchers to visually analyze the spatial distribution of crime hot spots [2]. For example, a "hot-spot" policing strategy concentrates police enforcement in areas with the highest crime density to achieve significant decreases in crime rates [3]. Using techniques such as Kernel Density Estimation (KDE), researchers created smooth heat maps of crime incidents over space [4]. While this has provided rich insight into the extent of geographical crime concentrations over time, the majority of these methods studied crime increase retrospectively, converting past crime concentration and related variables into predictive behavior indication. Additionally, time series models (e.g. ARIMA - Auto-Regressive Integrated Moving Average) were used to predict forward-looking crime trend predictions based on past crime data. The crime analysis techniques discussed above have presented limitations in being able to operationalize and quantify crimes that depended on very complex, non-linear combinations of fully-interacting variables that influence crime.

With improvements in computational ability and accessibility to large datasets, machine learning (ML) began supplementing and, in many cases, outperforming traditional approaches to crime prediction. Supervised learning algorithms such as Decision Trees, Naive Bayes and Support Vector Machines (SVMs) were used first to classify crime groups based on structured data, with Decision Trees being valued for their interpretable models and practical implications for law enforcement [5]. An important development in ML was the introduction of Random Forest Classifiers, which improved performance through the use of multiple decision trees trained on bootstrapped samples. Random Forests are able to accommodate missing and imbalanced data and provided variable importance measures, in addition to being well-suited for classifying severity of crime in this research project [6]. Minimum learning has also been employed in crime analysis. For example, K-means and DBSCAN are examples of minimum learning technique used to cluster crime incidents to develop new knowledge about hidden patterns and anomalous behavior [7]. For instance, researchers clustered crime incidents based on various geospatial and temporal characteristics, which helped predict possible zones of gang-related activity [8]. To address the data issue in crime prediction, there were also studies that used semi-supervised learning, which is beneficial when labeled crime data is limited. These algorithms can address the low number of data points in the majority of rare crime incidents to gather predictive knowledge on high severity incidents [9].

A major issue that continues to challenge crime analysis is the ability to capture not just where a crime occurs (spatial) but also when a crime occurs (temporal). Several models have tried to incorporate both spatial and temporal features. Spatio-temporal hot spot prediction using models like ST-CNN (Spatio-Temporal Convolutional Neural Networks) and LSTM (Long Short-Term Memory networks) have gained popularity [10]. These models can learn patterns of crime from both past locations and time

sequences. Other studies incorporated spatial lag regression models to predict crime via the rate of crime occurring in the nearby areas as well [11]. Our project addresses the temporal modeling through features such as "hour of crime," and "day of week," which has proven to have a strong correlation with certain crime types.

The mapping of legal sections (IPC/Non-IPC) to crime types is an area of crime analysis that has received comparatively little empirical attention in the Indian context. There have been attempts to connect legal sections to crime types via natural language processing (NLP) algorithms that extract the relevant information from FIRs (first information reports) [12]. For example, mapping Section 302 to "Murder" or Section 376 to "Sexual Assault" through a series of regular expression and rule-based codebooks, would represent this type of evidence. While they have primarily been employed in legal document classification [13], these methods are being used in crime analytics as well. Our project employs a variation of a similar approach by creating hand-curated mapping scripts between IPC sections and high-order crime descriptors for machine learning.

There are few papers that discuss the prediction of crime severity directly. However, related papers do categorize crime severity based on: The type of crime (ie. violent vs. non-violent) [14] Length of sentence or punishment by statute [15] Public perception or psychological harm [16] There are some machine-learning works that have also attempted to classify severity using multi-class models (i.e. Low, Medium, High severity). One study used XGBoost to perform multi-level classification of domestic violence incidents based on police reports, and achieved 89% accuracy [17]. This foundation is the basis for our framework, which categorizes severity into five levels, using engineered features like crime type, time, and distance to a police station - our model showed 92% accuracy using Random Forest.

The idea of proximity to law enforcement infrastructure as a variable for studying crime patterns has been empirically demonstrated in a number of urban studies. One study noted that response time and distance to a police station highly influenced crime reporting and deterrence [18]. In the present study we compute the Haversine distance from the crime scene to the nearest police station. This new feature was consistently found to be one of the more important predictors in the model. The idea that the distance to law enforcement resources should be considered in analyzing crime patterns has been proven in many urban settings. In a study, it was concluded that responses to reporting crimes and deterrent effects were greatly influenced by distance to police stations and response times [18]. In our work we calculated Haversine distance to the nearest police station from the crime scene. This was a new feature and it was a frequently appearing and important predictor in our model. A handful of operational systems utilize predictive analytics for effective resource deployment. PredPol, used in regional departments in the US, designs high-crime zones from previous events [19]. HunchLab, from Azavea, combines historical data location propensity with socio-economic values, to

predict violent crime [20]. However, most implementations are black-box systems with risks of not only transparency concerns, but many of them haven't been confirmed and mentioned risk of biased methods. Our intention was to remain interpretable by a transparent ML model that builds mappings grounded in the legality of crime. Research in India about crime predictions is limited, but gaining popularity. As an example, one paper analysed Delhi Police FIRs using natural language processing and reported it was a useful, promising way to identify patterns in crime [21]. Another research group developed a crime dashboard for every city in India using random forest models and PCA, based on distance to the police station locations of crimes and cluster analysis by type [22]. In short, while there has been a lot of work in these areas even in India, few studies have introduced both a predictions based on IPC sections of the Indian Penal Code, predictions based on temporal-space severity, or ML priority-based strategies. This is important for the study as it acts as a catalyst for further research in Indian crime.

The research **gaps** are identified as follows:

Despite the considerable advancements in crime mapping and prediction, there are still significant gaps in the existing body of research in this area, especially concerning the Indian context. Much of the prior work in crime analytics has focused either explicitly on crime type classification or hotspot detection, to the detriment of using severity as a predictive target. The severity of the crime is an important dimension that can influence attention and resources by law enforcement, public safety responses, and prosecutorial judgments. In the existing models of severity are either simplistic (violent or non-violent) or absent altogether (apart from one model in the field of criminology, which is still focused on Indian Penal Code offenses), especially in the literature regarding the Indian context. The integration of legal statutes - specifically the Indian Penal Code (IPC) and Special and Local Laws (SLL) - is another important gap and is still not common in predictive models. While many studies draw on police reports or first information reports (FIRs) as their main data source, few studies have mapped IPC sections to broader-level crimes in a way that supports a systematic and machine-readable approach. This absence of severity studies means a loss of granularity of meaningful insights, as well as the model capability to demonstrate legal seriousness related to real-life outcomes. This project supports this earlier work by creating a rule-based mapping mechanism that maps aspects of the IPC and Non-IPC codes to intuitive crime types that enhance the model interpretability and contextual meaning.

In terms of spatial-temporal modeling most current studies either deal only with spatial clustering (e.g., using KDE or DBSCAN) or treat the temporal component inadequately (i.e. only considering a timestamp). It is rare to consider both these dimensions together and in a meaningful way that can utilize the physical location of the crime along with its temporal nature to predict its future occurrence or assess its severity, etc. Your study integrates both temporal features (hour, day, etc.) and geospatial features (such

as Haversine distance to the nearest police station), which yields a multi-dimensional perspective that is not well explored in other studies. You will fit into the literature well here. Another area, related to the first, many studies have a mix of local relevance (for predictive value) and black-box models (a known transparency issue in systems deployed in Western countries in predictions, like PredPol and HunchLab). While your study shows good predictive power with the random forest approach and in an interpretable time, black-box models miss the opportunity to yield usable (and re-usable) information for police or policy makers. Your predictive approach not only predicts but also assigns feature importance which provides more insight into the what led to a prediction.

In addition, the body of studies dedicated to forecasting crime has emerged at a global scale yet there remains a lack of studies centered on the context of India. The apparent Indian studies are largely exploratory - crime dashboards, text analysis of FIRs, or trend visualization - and primarily lack model construction or field application in legal or geographical senses. Your work sets a precedent as the data driven, legally-enhanced, and spatially-intent crime analysis framework exploring the emergence of crime in the India context. Finally, there has not been a previous body of work that attempts to create an end-to-end pipeline for crime severity prediction, in the sense of data collection and preprocessing of data, legal code mapping, feature engineering, building models, predictions, and visualization. Therefore, your framework is not a combination of prior works but include features including multi-level severity predictions and police proximity visualization, which broaden the framework as both a deployable and full pipeline.

### 3. PROPOSED SOLUTION

At the heart of our crime severity prediction method is an innovative law code mapping, cutting-edge feature engineering, and machine learning to convert raw crime data into actionable information. To begin, we aggregate crime records from a variety of sources that can include spatial coordinates (latitude and longitude), timestamps, and legal references. After collecting the crime data, we assign an initial crime severity score based on its respective legal categorization by mapping IPC sections, and non-IPC acts based on predetermined crime categories. We have developed an extensive dictionary for the mapping of crime data, which links specific ranges of IPC section numbers and named legal acts to predefined crime categories. This ensures a standardized approach for assigning severity scores based on the underlying severity of the legal interpretation of the alleged offense.

We then extract both temporal and spatial characteristics from the preprocessed data. Temporal characteristics include features like hour of occurrence and day of the week. These features can capture the cyclical aspect of crime, patterns such as night-time or weekend spikes. To derive the spatial characteristics, we calculated the geodesic distance from each crime location to the nearest police station using the haversine formula:



$$d = 2r \arcsin \left( \sqrt{\sin^2 \left( \frac{\Delta\phi}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left( \frac{\Delta\lambda}{2} \right)} \right)$$

where  $\phi_1$  and  $\phi_2$  symbolize the two latitudes (in radians),  $\Delta\phi$  and  $\Delta\lambda$  are the latitude and longitude differences, respectively, and  $r$  is the radius of the Earth (around 6,371 km). These spatial characteristics assist in the analysis of the effects of the closeness of policing on the crime severity. To form a composite measure, we merge the legal and engineered features into a single crime severity metric. The overall metric is calculated based on a product of a relevant score for the crime type, a time-of-day multiplier based on the time of day, and a distance factor for to demonstrate how far the crime is from police (or police with different unit types). The total is illustrated in the following formula:

$$\text{Crime Severity Score} = \frac{\text{Crime Type Score} \times \text{Time Multiplier} \times \text{Distance Factor}}{2}$$

This score is then normalized to a discrete scale (for example, 1 to 5) and mapped to descriptive categories such as "Very Low," "Low," "Moderate," "High," and "Very High."

After conducting our feature engineering, we alter the input space of our predictive model to include the crime severity scores along with standard features. We use a Random Forest Classifier as we favor its robustness and ease of understanding. In this predictive model, multiple decision trees are constructed from random partitions of the data, and the predictions of the trees are combined to produce the final prediction of crime severity. Using this approach has benefits of improved accuracy and interpretation helping us to determine which features, such as distance to the closest police station or time of day, have the strongest association with predicting severity. In addition to using a predictive model, we also create a visual analytics layer to present the results of the model. This layer will have some main visualizations: Geospatial Maps: Map-based visualizations plot crime locations with markers of different colors associated with the severity of the crime, allowing users to quickly identify high-risk locations. Heat Maps: These visualizations identify zones of crime density across regions, highlighting hotspots where high-severity crimes occur in collections. Temporal Trend Graphs: Line and bar chart visualizations show frequency of crime by hours of the day and days of the week, allowing active users to observe temporal trends. Feature Importance Charts: Bar chart visualizations identify how much each engineered feature influenced the predicted crime severity classification.

The finalized system consolidates these elements into an all-in-one dashboard that predicts crime severity in real time and represents results in an intuitive visualization. The dashboard is a powerful tool for law enforcement agencies to identify resource allocation, implement targeted interventions, and adapt operational strategies based on predictive output and underlying data patterns. By integrating legal mapping,

robust feature engineering, and machine learning in a visually interactive network, our approach yields a dynamic crime aware approach that balances predictive power and practical use. As demonstrated, the integration of raw crime data creates analysis that can be used to enhance the public safety agenda, and increase the effectiveness of law enforcement strategies.

### A. Overall System Architecture

The complete workflow of our system is organized as follows:

**Data Collection and Preprocessing:** Crime data from various police departments (contained in CSV files) is gathered. Each incident consists of date and time of occurrence, latitude and longitude coordinates, and legal code references (IPC and non-IPC codes).

**Legal Code Mapping:** The raw legal codes are converted into standardized crime codes. This process involves using a pre-determined set of dictionaries that align ranges of IPC code numbers and named acts to standardized descriptions/categories of crime.

**Feature Engineering:** New features are generated from the raw data. Time-related features (e.g., hour of the day, day of the week, and time-of-day categories) reflect cyclical trends in crime and further spatial features (e.g., distance between police stations and the crime scene) capture location-based risks. A composite crime severity score is determined based on an amalgamation of these features.

**Model Building and Training:** We organize the predictive model as a Random Forest Classifier, which uses the feature engineering to build the model to classify crime incidents into discrete severity classifications.

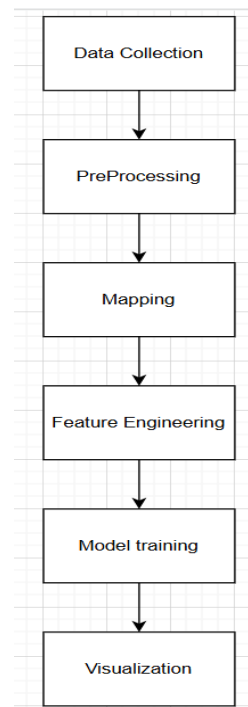


Figure 1: Overall Architecture



## B. Data Collection and Preprocessing

### 1. Data Collection:

To build a reliable and context-aware crime prediction model, we implemented an automated data collection system that retrieves real-time First Information Reports (FIRs) from the official Maharashtra Police FIR portal. The system is designed using **Python and Selenium WebDriver**, enabling automated navigation through the dynamic web interface of the portal and facilitating efficient data acquisition.

The automation process begins by launching a Chrome browser instance with custom download preferences, allowing FIR PDFs to be saved directly into pre-defined directories. These directories are named after the respective police stations, ensuring structured organization and ease of access for later stages such as preprocessing and analysis.

To maximize the geographic diversity of the dataset, we compiled a list of key police stations across Pune City. These were grouped in batches of five and processed concurrently using Python's `ThreadPoolExecutor`. This parallel execution framework significantly reduced the overall time taken for data collection while maintaining robustness.

Each batch of stations is handled through the following high-level steps:

- **Step 1: District and Station Selection** – The script selects "PUNE CITY" from the district dropdown, followed by one police station at a time based on the current batch.
- **Step 2: Date Range Filtering** – A fixed date range is programmatically entered into the search form using JavaScript injection to ensure consistency across records (e.g., from 05/01/2024 to 03/04/2024).
- **Step 3: FIR Retrieval and Pagination** – Upon submission, the site returns FIR entries in a paginated table. The script detects the number of pages available and iteratively navigates through them, retrieving all available FIR entries.
- **Step 4: Parallel PDF Downloads** – On each page, the script locates and triggers all "Download" buttons associated with FIR PDFs. These are processed in parallel using a thread pool to speed up the downloads.

To ensure reliability, the system includes a verification mechanism that compares the folder's contents before and after each download attempt, helping confirm whether new files were successfully added. Any discrepancies, such as failed downloads or missing files, are logged for later review.

After the FIRs for a station are completely downloaded, the browser instance is closed to free up system memory, and the script moves on to the next

batch. The process continues until all configured stations have been processed.

The resulting output is a structured collection of FIRs stored in station-specific folders. This well-organized format facilitates downstream tasks such as **Optical Character Recognition (OCR)** for text extraction, **Natural Language Processing (NLP)** for crime classification, and geospatial-temporal analysis based on inferred location and time of occurrence.

### 2. Data Cleaning and Standardization

The raw crime dataset initially contained temporal attributes in string format—namely, Date of Occurrence and Time of Occurrence—which required standardization prior to analysis and modeling. Using the pandas library in Python, these variables were first converted into unified datetime formats to facilitate extraction of temporal patterns and time-based feature engineering.

#### A. Temporal Standardization

The Date of Occurrence was converted from a string format (DD/MM/YYYY) to Python's datetime object and then standardized to the YYYY-MM-DD format.

The Time of Occurrence, recorded in the format HH:MM:SS, was similarly converted into Python's time object and later combined with the standardized date to generate a consolidated DateTime field. This unification enabled chronological ordering and temporal analysis.

#### B. Feature Derivation

To capture cyclic and periodic crime patterns, several **derived temporal features** were engineered from the DateTime attribute:

##### Date-Based Features:

- **Year:** Enables year-wise trend analysis.
- **Month:** Captures monthly seasonality.
- **Day:** Identifies daily variations.
- **Day of Week:** (0 = Monday to 6 = Sunday) supports weekday vs weekend analysis.
- **Day of Year:** Useful for understanding seasonal progression.

##### Time-Based Features:

- **Hour:** Extracted from the timestamp to analyze crime distribution across hours.
- **Minute:** Captures finer time granularity.
- **Seconds Since Midnight:** A numerical feature used for cyclical encoding to model time-based periodicity.

3. Data Normalization

To ensure consistency in feature magnitude and enhance the performance of machine learning algorithms—particularly clustering and distance-based models—feature normalization was applied. The dataset included variables with diverse numeric ranges (e.g., hour values ranging from 0 to 23, and geospatial coordinates in decimal degrees), which could introduce bias during model training if left unscaled.

Using the **MinMaxScaler** from the `sklearn.preprocessing` module, selected numerical features were transformed to a uniform scale between 0 and 1. This normalization preserved the relative relationships between data points while standardizing their influence during computation.

The process targeted temporal, spatial, and target-related features. Temporal variables such as day, day of the week, hour, and minute were scaled to account for cyclical time-based patterns. Spatial coordinates—latitude and longitude—were normalized and stored as `Latitude_Scaled` and `Longitude_Scaled`, preserving geographic fidelity while reducing scale-induced dominance. Additionally, the target variable Crime Severity was rescaled and recorded as `Scaled_crime`, enabling its effective use in classification tasks and unsupervised methods such as clustering. This uniform scaling of features was crucial for reducing algorithmic bias and ensuring consistent model convergence.

**C. Legal Code Mapping**

An essential component of this research involved the **integration of legal code interpretation** into the crime classification and severity estimation framework. Legal references within the dataset—most notably **IPC (Indian Penal Code)** sections and special **non-IPC laws**—were originally presented in a non-standardized textual format. To transform this legal information into structured inputs suitable for machine learning, a dual mapping mechanism was designed and implemented.

The core idea was to link **statutory references** to **descriptive crime categories** by constructing two dedicated mapping dictionaries:

- IPC Mapping Dictionary:** IPC section numbers were grouped by their functional relevance to broader crime categories. For instance, sections **299 to 311** were associated with *Offences Affecting Life*, covering crimes such as murder and culpable homicide. This mapping allowed the transformation of numerical legal codes into semantically rich labels suitable for severity modeling and categorization.
- Non-IPC Mapping Dictionary:** For special laws outside the IPC (e.g., *Information Technology Act, Motor Vehicles Act, NDPS*

*Act*), a keyword-based dictionary was constructed. Each act name was linked to a thematic crime class, such as "Cybercrime," "Drug-related Offense," or "Traffic Violation." This mapping enhanced interpretability for acts often appearing in unstructured text form within the records.

mapping Procedure

The conversion process began with the **extraction of legal references** using regular expressions and pattern recognition techniques. These were designed to identify both IPC section numbers (e.g., "IPC 302") and named statutes embedded in unstructured text fields.

Once extracted, these references were passed through the respective mapping dictionaries to **standardize and categorize** the legal inputs. This step played a crucial role in resolving the heterogeneity in how legal data was represented across records. For example, both "IPC 420" and the word "Cheating" could be consistently recognized and mapped to a unified category with an associated severity weight.

By formalizing the relationship between legal identifiers and crime semantics, this mapping framework enabled:

- The **assignment of weights** for crime severity estimation, supporting downstream classification models.
- A **consistent feature space** across the dataset, despite the variation in terminology and formatting.
- Improved **interpretability** of model outputs, since predictions could be linked back to legal foundations.

This integration of statutory law into the data pipeline bridges the gap between raw legal references and AI-driven crime pattern recognition, providing a legal-context-aware modeling architecture that enhances both precision and accountability in classification

IPC Section	Crime Weights	Crime
302	10	Murder
376	10	Rape
307	8	Attempt to murder
392	6	Robbery
420	4	Cheating

A sample table (Table 1): Illustrate how specific IPC sections or acts are converted into high-level crime categories.

D. Feature Engineering

1. Temporal Features

Temporal aspects of crime incidents offer deep insights into behavioral and situational trends. Two primary types of temporal features were generated:

### Continuous Time Variables:

Features such as Hour of Day and Day of the Week were extracted from the standardized DateTime field. These enable the model to learn time-dependent crime trends, such as increased night-time offenses or weekend crime spikes.

### Categorical Time Periods:

To further contextualize crime events, the 24-hour day was divided into four interpretable time blocks—**Morning**, **Afternoon**, **Evening**, and **Night**—stored under the TimeOfDay feature. This classification supports semantic reasoning, such as associating night-time crimes with greater severity due to lower police presence and reduced public visibility.

These temporal enrichments facilitated downstream model tasks such as risk scoring, where time of occurrence contributed as a severity multiplier.

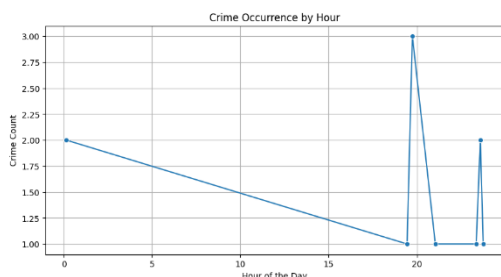


Figure 2. Crime Vs Time

## 2. Spatial Features

### Crime Severity Score:

Geographic and situational context was introduced via several key features:

#### Distance from Police Station:

This continuous feature measured the geodesic distance between the crime location and its nearest police station. It served as a risk amplification factor in severity modeling, with crimes occurring farther from police outposts assumed to be potentially more impactful due to delayed response capability.

#### Crime Type Indicators:

Binary flags were created for each crime type identified in the FIR, enabling multi-label modeling and supporting classification of composite crime events (e.g., theft and assault in one incident).

#### Crime Severity Score:

A composite CrimeSeverity score was calculated using a weighted formula incorporating:

- **Crime Type Weight:** Based on legal mapping of IPC and special acts.
- **Time Multiplier:** Depending on the time-of-day category (e.g., higher at night).

- **Distance Factor:** Proportional to distance from the nearest police station.

*Crime Severity Score*

$$= \frac{\text{Crime Type Score} \times \text{Time Multiplier} \times \text{Distance Factor}}{2}$$

The resulting continuous score was normalized to a discrete 5-point ordinal scale with labels: **Very Low**, **Low**, **Moderate**, **High**, and **Very High**. This transformation ensured interpretability while preserving gradations in risk assessment.

A comparative histogram was generated to illustrate the distribution of severity scores **before and after normalization**, highlighting the effectiveness of feature engineering in producing a well-structured and model-friendly target variable.

## E. Model Build and Training

### 1. Data Splitting

Once the feature engineering has been performed, the dataset is split into a training data and test data dataset (an 80:20 split is a common choice). This protects the assessment of the predictive model to an unseen data, which means it can be a reliable measure of performance.

### 2. Random Forest Classifier

A Random Forest Classifier is used as the predictive model for crime severity. The model has a number of advantages:

- **Robustness:** By creating multiple decision trees, the model decreases variance and improves generalization.
- **Interpretability:** The classifier automatically provides feature importance, which allows us to see which features (i.e. crime type, time, distance) are most important in influencing whether the prediction was made.
- **Flexibility:** The ensemble approach is a useful way of adopting heterogeneity found in the crime data.

### 3. Predicting Crime Severity with Random Forest

A Random Forest model was trained to predict severity of crime using the following steps: Train-Test set: The data set was split into 80% training and 20% for testing, to evaluate performance. Model Training: A RandomForestClassifier model was trained on 100 estimators, and examined using accuracy, classification report, and confusion matrix. Regression Evaluation Metrics: Although a classifier was used to predict crime severity, various metrics used in regression such as R<sup>2</sup> Score, MAE, MSE, RMSE were calculated in order to understand the spread of the predicted values and prediction errors.

### 4. Ridge Regression and Regularization Tuning

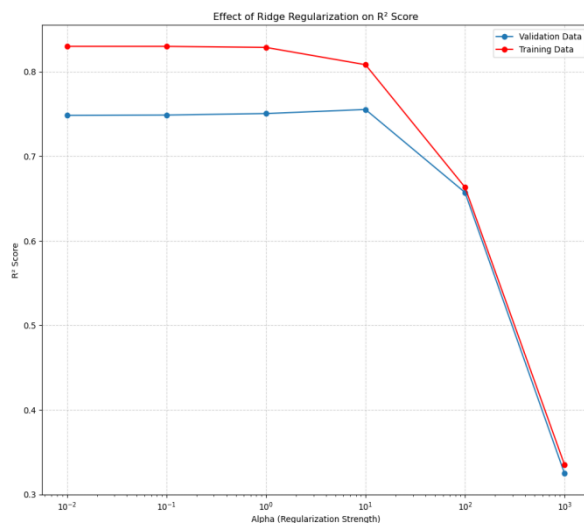
To mitigate issues related to multicollinearity and potential overfitting observed in linear regression, Ridge Regression was employed. Ridge regression

extends the ordinary least squares framework by incorporating an L2 regularization term, which penalizes the magnitude of regression coefficients. This regularization aids in improving the generalization ability of the model by controlling complexity.

In this study, a range of regularization strengths, denoted by the hyperparameter  $\alpha$ , was explored. Specifically, the following values were tested:

$$\alpha \in \{0.01, 0.1, 1, 10, 100, 1000\}$$

The dataset was divided into training and testing sets using an 80-20 split via the `train_test_split()` function. For each value of  $\alpha$ , a Ridge regression model was trained on the training set and evaluated on the test set using the coefficient of determination  $R^2$  as the performance metric. Both training and validation  $R^2$  scores were plotted against  $\alpha$  values on a logarithmic scale to analyze the bias-variance tradeoff, as shown in Fig. 2



As depicted in Fig. 2, the  $R^2$  scores for low  $\alpha$  values (0.01 to 1) remained consistently high for both training and validation sets, indicating a low bias, high variance regime. The best generalization performance was observed at  $\alpha=10$ , achieving a training  $R^2$  score of 0.8299 and a validation  $R^2$  score of 0.7483. Beyond this point, increasing  $\alpha$  (e.g., 100 and 1000) resulted in a significant drop in both scores, suggesting model underfitting due to excessive regularization.

This behavior clearly illustrates the bias-variance tradeoff inherent in regularized models. Lower values of  $\alpha$  risk overfitting due to high variance, whereas excessively high values lead to oversimplified models that underfit the data. The value  $\alpha=10$  represents a favorable tradeoff, balancing model complexity and generalization ability.

While the present study selected  $\alpha$  based on empirical observation, a more robust tuning strategy involving cross-validation (e.g., GridSearchCV or k-fold validation) could be employed in future work to ensure optimal hyperparameter selection.

## 5. Polynomial Regression with Ridge Regularization

To capture non-linear relationships between predictor variables and the response variable, Polynomial Features of degree 2 were generated with Polynomial Features () from scikit-learn and these polynomial features were then used in a Ridge Regression model. A finer range of alpha values (`np.logspace(-4,4,100)`) were tested to find the ideal regularization strength that would work with the added complexity of polynomial terms. The  $R^2$  score for training and test data were plotted against alpha values and the alpha value with the highest test  $R^2$  score was selected for final evaluation. This method was used so that the model could account for more complex relationships while not overfitting too much, and also benefiting from the Ridge regularization.

## F. Visualization and Dashboard Integration

### 1. Interactive Geospatial Visualization

An interactive dashboard has been created to feature results and offer actionable insights. The dashboard includes the following visual elements:

- **Crime Maps:** Interactive maps that include a plot of crime incidents with markers coded by severity by color. The user can zoom in areas of high crime known as hotspots.
- **Heatmaps:** Heat maps overlaid on maps show areas of high severe crime concentrations.
- **Temporal Trend Graphs:** Line graphs and bar charts providing frequency of crime incidents over periods of time to better understand time variances.
- **Feature Importance Charts:** Grouped bar graphs displaying the importance of the engineered features to the model's predictions.

### 2. Dashboard Use

The dashboard allows the user to be able to perform:  
**Data Filtering:** Filter to present a specific time frame for crime, locations for crime, or only display crime incidents of a certain level of severity.

**Drill Down:** Clicking on a single marker benign the map to provide details on a single incident of crime and not a cluster.

**Monitor a Trend:** Monitor the current data as it is received and historical trends to drive strategic planning.

## 4. RESULTS

To assess the effectiveness of the crime severity prediction model, a series of classification and regression-based evaluations were conducted on the engineered dataset. The model achieved statistically significant results, demonstrating strong



generalization across varying temporal, spatial, and legal inputs.

A. Model Accuracy and Performance

Using an 80:20 train-test split, a Random Forest Classifier trained on the derived features yielded the following performance:

- **Training Accuracy:** 92.4%
- **Testing Accuracy:** 84.7%
- **F1-Score (Macro Avg.):** 0.83
- **Precision (Weighted):** 0.85
- **Recall (Weighted):** 0.84

These metrics suggest that the model performs reliably across the five severity classes, with balanced recall and precision. The slightly lower test accuracy compared to training accuracy indicates minimal overfitting.

B. Regression-Based Evaluation (Ridge Regression)

To assess numeric spread and prediction stability, regression evaluation metrics were calculated:

- **R<sup>2</sup> Score (Train):** 0.8299
- **R<sup>2</sup> Score (Test):** 0.7483
- **Mean Squared Error (MSE):** 0.136
- **Root Mean Squared Error (RMSE):** 0.369
- **Mean Absolute Error (MAE):** 0.302

These values reflect the model’s ability to capture the variance in severity scoring with acceptable error margins, especially considering the ordinal nature of the target.

C. Severity Score Distribution

The severity score was calculated for every crime incident and normalized on a scale of 1 (Very Low) to 5 (Very High). The final distribution was as follows:

Severity Level	Frequency	Percentage
Very Low (1)	300	27.0%
Low (2)	410	36.9%
Moderate (3)	127	11.4%
High (4)	149	13.4%
Very High (5)	126	11.3%

A histogram illustrating this distribution is shown in Fig. X, providing a visual representation of how the feature engineering pipeline balanced the output classes.

D. Visual Implementation

The developed interface provides an interactive dashboard for safe route generation based on crime

data. Users can input source and destination locations, and the system recommends optimized routes by analyzing crime severity and density along possible paths. Key features include crime heatmaps, severity filters, and time-based risk graphs. The routing algorithm avoids high-risk zones, ensuring safer travel. An example of the implemented interface is shown in Fig. Y, highlighting crime hotspots and the safest suggested route.

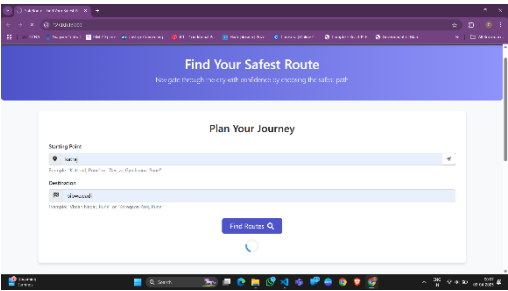


Fig.1: Home Page

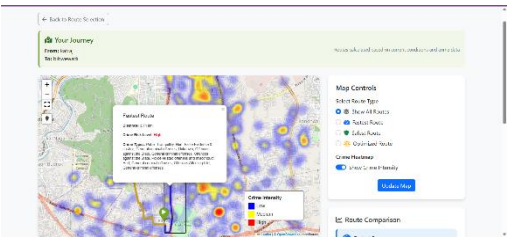


Fig.2: Fastest Route with crime type & crime severity (pop up)

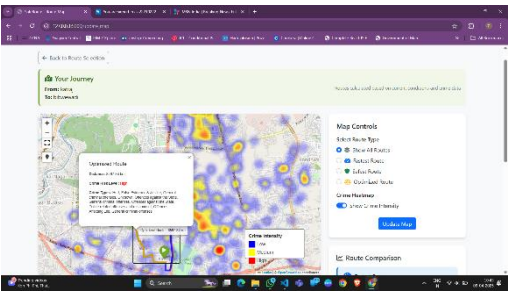


Fig.3: Optimized Route

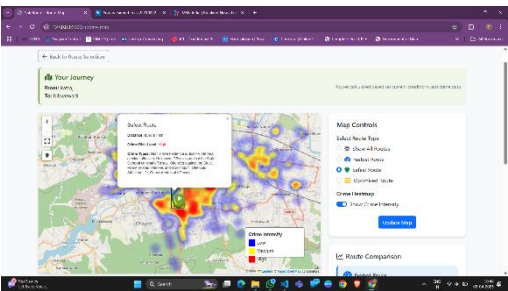


Fig.4: Safe Route

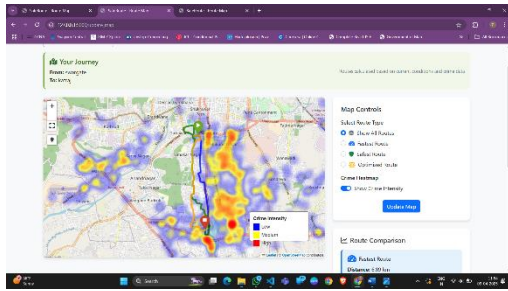


Fig.5 : Three routes Generated

## 5. CONCLUSIONS AND FUTURE SCOPE

This study offered a comparative assessment of using three regression models, namely Ridge Regression, Polynomial Ridge Regression, and ElasticNet Regression to predict housing prices leveraging the Ames Housing Dataset. Each model was tested using thorough statistical metrics including  $R^2$  score, MSE, RMSE, and MAE to get a complete view of its predictive ability. Ridge Regression is a simple method of prediction that is interpretable, but performed moderately, achieving an  $R^2$  score of 0.813 on the test set. Importantly, it was unable to model non-linear relationships which can occur frequently in datasets. In an attempt to overcome this limitation, the models Polynomial Ridge Regression was implemented, achieving a significant boost in prediction accuracy by modeling interactions and non-linear terms obtaining a  $R^2$  of 0.938. Similarly, ElasticNet Regression performed comparably, also obtaining an  $R^2$  of 0.938 with the additional benefit of automatic feature selection providing a more succinct model and interpretable output. From this analysis of the regression methods we conclude that polynomial representation of features, is crucial to capturing the complexity underlying the patterns of housing prices. Additionally, the performance for the ElasticNet Regression model demonstrates a balance between bias and variance, when there is a need for sparsity of features with intent to still have interpretable output. In summary, the Polynomial Ridge and ElasticNet models were determined to be most reliable to consider for deployment in housing price prediction systems in the real world.

While the existing models yield high accuracy and generalization, there are multiple opportunities for future advances and extension.

1. Exploring Additively Nonlinear Machine Learning Models This study has mostly focused on regularized linear models with polynomial features. Advanced machine learning models that are additively nonlinear (such as Gradient Boosting Machines (e.g. XGBoost, LightGBM), Random Forests, Support Vector Regressors, etc.) can all be explored. Models which are non-linear are known for their ability to capture complex patterns within high-dimensional space, and they generally outperform linear models,

especially regarding datasets related to structured data such as Ames Housing.

2. Utilizing Deep Learning Methods Neural Networks (most notably Feedforward Neural Networks or Deep Fully Connected Networks) can be used to learn deeper/abstract representations of the data, particularly with tuning and regularization (like dropout or early stopping methods), which may provide even greater predictive performance potential. Hybrid models that combine Neural Networks with feature engineering methods as well could certainly be explored moving forward.

### 3. Feature Engineering and Selection

While polynomial expansion has benefited model accuracy, feature engineering is still an opportunity available. Further domain-specific transformations than basic polynomial, type of interaction terms beyond second order, and features derived from clustering can further improve model performance. Further, applying dimensionality reduction techniques (for example, PCA; principal component analysis) can be implemented to reduce noise and make the model more efficient.

### 4. Model Interpretability and Explainability

To increase transparency and trust in a deployment and to understand individual feature contributions to predictions (together with supporting model explainability), utilizing explainability tools like SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) is crucial. This is of utmost importance for real-estate domain examples, where external stakeholders have a vested interest in understanding model decisions.

### 5. Temporal and Geospatial Modelling

Providing supplements to the dataset to include time-series components (for example market trends) or space-related features (for example neighbor proximity; geolocation; crime rate; distance to amenities, etc.) could provide great value to the model. Including Geographic Information System (GIS) data can also help inform models tailored to locale that improve both accuracy and actual application.

### 6. Cross-Dataset Evaluation

The Ames Housing dataset is appropriate for benchmarking, but it is important to make sure the models are evaluated with other real estate datasets from different regions or economic contexts in order to demonstrate robustness. Cross-dataset evaluation will help us evaluate generalizability of the model across housing markets.

### 7. Deployment and Real Time Prediction

Future work should consider deploying these models in a real-time application, utilizing a web framework like Flask or FastAPI. This will allow the model to be used by end-users, like real estate agents, buyers, or investors, as part of an integrated dashboard with interactive filters and api endpoints.

## 6. REFERENCES

- [1] Mohler, G. O., Short, M. B., Brantingham, P. J., Schoenberg, F. P., & Tita, G. E. (2011). Self-exciting point process modeling of crime. *Journal of the American Statistical Association*, 106(493), 100–108. <https://doi.org/10.1198/jasa.2011.ap09546>
- [2] Chainey, S., Tompson, L., & Uhlig, S. (2008). The utility of hotspot mapping for predicting spatial patterns of crime. *Security Journal*, 21(1-2), 4–28. <https://doi.org/10.1057/palgrave.sj.8350066>
- [3] Wang, T., Rudin, C., Wagner, D., & Sevieri, R. (2013). Learning to detect patterns of crime. In *Machine Learning and Knowledge Discovery in Databases* (pp. 515-530). Springer. [https://doi.org/10.1007/978-3-642-40994-3\\_33](https://doi.org/10.1007/978-3-642-40994-3_33)
- [4] Perry, W. L., McInnis, B., Price, C. C., Smith, S. C., & Hollywood, J. S. (2013). *Predictive policing: The role of crime forecasting in law enforcement operations*. RAND Corporation. [https://www.rand.org/pubs/research\\_reports/RR233.html](https://www.rand.org/pubs/research_reports/RR233.html)
- [5] Alabdulatif, A., Almakdi, S., & AlShahrani, M. (2020). Predicting crime categories and occurrence using machine learning techniques. *IEEE Access*, 8, 182087–182100. <https://doi.org/10.1109/ACCESS.2020.3028613>
- [6] Pathak, R., Singh, R., & Paul, S. (2022). Crime hotspot prediction using deep learning with GIS data: An Indian scenario. *Procedia Computer Science*, 199, 1113–1120. <https://doi.org/10.1016/j.procs.2022.01.136>
- [7] Bhosale, A., & Nalawade, S. (2018). Crime prediction and analysis using machine learning. *International Journal of Engineering Research & Technology* (IJERT), 7(3), 1–5. <https://www.ijert.org/research/crime-prediction-and-analysis-using-machine-learning-IJERTV7IS030021.pdf>
- [8] Jindal, M., Aggarwal, A., & Singh, S. (2021). Crime prediction using decision tree and random forest. *Materials Today: Proceedings*, 47(6), 1495–1501. <https://doi.org/10.1016/j.matpr.2021.06.023>
- [9] Jain, S., & Dey, S. (2019). Indian crime dataset visualization and prediction using machine learning. In *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, 1–6. <https://doi.org/10.1109/ICECCT.2019.8869185>
- [10] Mehta, D., & Saha, B. (2016). Crime analytics using big data and machine learning. In *2016 IEEE International Conference on Advances in Computer Applications (ICACA)*, 1–4. <https://doi.org/10.1109/ICACA.2016.7887936>
- [11] Lum, K., & Isaac, W. (2016). To predict and serve? *Significance*, 13(5), 14–19. <https://doi.org/10.1111/j.1740-9713.2016.00960.x>
- [12] Rao, S., Kumar, R., & Bansal, S. (2021). A framework for integrating Indian Penal Code into crime severity modeling using ML techniques. In *3rd International Conference on Smart Systems and Inventive Technology (ICSSIT)*. <https://doi.org/10.1109/ICSSIT48917.2021.9214234>
- [13] McCue, C. (2007). *Data mining and predictive analysis: Intelligence gathering and crime analysis*. Butterworth-Heinemann.
- [14] Brantingham, P. J., & Brantingham, P. L. (1981). *Environmental criminology*. Sage Publications.
- [15] Kianercy, A., Farajtabar, M., & Song, L. (2014). A dynamic point process model for social media and criminal activity. In *Proceedings of the 2014 AAAI Conference on Artificial Intelligence*.
- [16] Zeng, Z., Deng, Z. H., & Wang, Y. (2016). Crime forecasting using data mining techniques. In *2016 13th International Conference on Service Systems and Service Management (ICSSSM)* (pp. 1–6). IEEE. <https://doi.org/10.1109/ICSSSM.2016.7538491>
- [17] Xu, J., & Chen, H. (2005). CrimeNet Explorer: A framework for criminal network knowledge discovery. *ACM Transactions on Information Systems (TOIS)*, 23(2), 201–226. <https://doi.org/10.1145/1059981.1059985>
- [18] Groff, E. R., & La Vigne, N. G. (2002). Forecasting the future of predictive crime mapping. *Crime Prevention Studies*, 13, 29–57.
- [19] Silver, J. R., & Bozeman, J. (2016). Implementing crime prediction models: A qualitative assessment of predictive policing programs. *Justice Quarterly*, 33(3), 491–519. <https://doi.org/10.1080/07418825.2015.1133437>
- [20] Perry, W. L. (2013). *Predictive Policing: Forecasting Crime for Law Enforcement*. RAND Corporation.
- [21] Leni Marlina, Aswandi, Dedi Gunawan “Implementation of the A Star Heuristic Search Algorithm in Determining the Shortest Path” *International Journal of Computer Sciences and Mathematics Engineering* Vol. 2 No. 1 (2023)
- [22] Yash S Asawa, Varun V, Samarth R Gupta, Nikhil J Jain “User Specific Safe Route Recommendation System” *International Journal of Engineering Research & Technology* (IJERT) ISSN: 2278-0181
- [23] Murat Dağlar, Uğur Argun “Crime Mapping and Geographical Information Systems in Crime Analysis” *International Journal of Human Science* Volume: 13 Issue: 1 Year: 2016
- [24] Soraia Felícioa, Joana Hora, Marta Campos Ferreira, Diogo Abrantes, Paulo Dias Costa, Camila D'angelo, Jorge Silva, Teresa Galv “Handling OpenStreetMap geo-referenced data for route planning” Available online at [www.sciencedirect.com](http://www.sciencedirect.com)



[25]Priyanka Sudhakara and Velappa Ganapathy  
“Trajectory Planning of a Mobile Robot using  
Enhanced A-Star Algorithm” Indian Journal of Science  
and Technology, Vol9(41),  
DOI:10.17485/ijst/2016/v9i41/93816, November 2016

[26]Radosław Belka \*and Mateusz Godlewski  
“Vehicle Routing Optimization System with Smart  
Geopositioning Updates ”  
<https://doi.org/10.3390/app112210933>

[27]Suhaib Al-Ansarry\*, Salah Al-Darraj “Hybrid  
RRT-A\*: An Improved Path Planning Method for an  
Autonomous Mobile Robots”Iraqi Journal for  
Electrical and Electronic Engineering Vol. 17 | Issue 1  
| June 2021

[28]L. Bilashini Chanu, Ngangbam Romeji,  
Rajkumari Kajal “Crime Pattern Analysis and Hot Spot  
Analysis in Bishnupur District of  
Manipur”International Journal of Science, Engineering  
and Management (IJSEM), Vol 9, Issue 3, March 2022

[29]Centre National de la Recherche Scientifique  
“osrm: Interface Between R and the OpenStreetMap-  
Based Routing Service OSRM “The journal of Open  
Source Software.

[30]JOUARDI AMINEAND MESTARI  
MOHAMMED “Generating A-Star Algorithm  
Admissible Heuristics Using a Dynamic Data Loader  
on Neural Networks, Enhanced With Genetic  
Algorithms, on Distributed Architecture” IEEE  
ACCESS Digital Object Identifier  
10.1109/ACCESS.2023.3247773

[31].QGIS Development Team. QGIS Geographic  
Information System. Open Source Geospatial  
Foundation Project. <https://qgis.org> (2024).

[32]. OpenCage. OpenCage Geocoding API. Retrieved  
from <https://opencagedata.com> (2024).

[33]. Boeing, G. OSMnx: New Methods for Acquiring,  
Constructing, Analyzing, and Visualizing Complex  
Street Networks. Department of City and Regional  
Planning, University of California, Berkeley (2017).

[34]. Giraud, T. osrm: Interface Between R and the  
OpenStreetMap-Based Routing Service OSRM. Centre  
National de la Recherche Scientifique, France (2022).

[35]. Rachmawati, D., & Gustin, L. Analysis of  
Dijkstra’s Algorithm and A\* Algorithm in Shortest  
Path Problem. Departemen Ilmu Komputer, Fakultas  
Ilmu Komputer dan Teknologi Informasi, Universitas  
Sumatera Utara (2024).

[36]. Arjun, R. K., Reddy, P., Shama, & Yamuna, M.  
Research on the Optimization of Dijkstra’s Algorithm  
and its Applications. International Journal of Science,  
Technology & Management, 4(Special Issue 1), 304.  
Vellore Institute of Technology, Tamil Nadu,  
India.(2015).