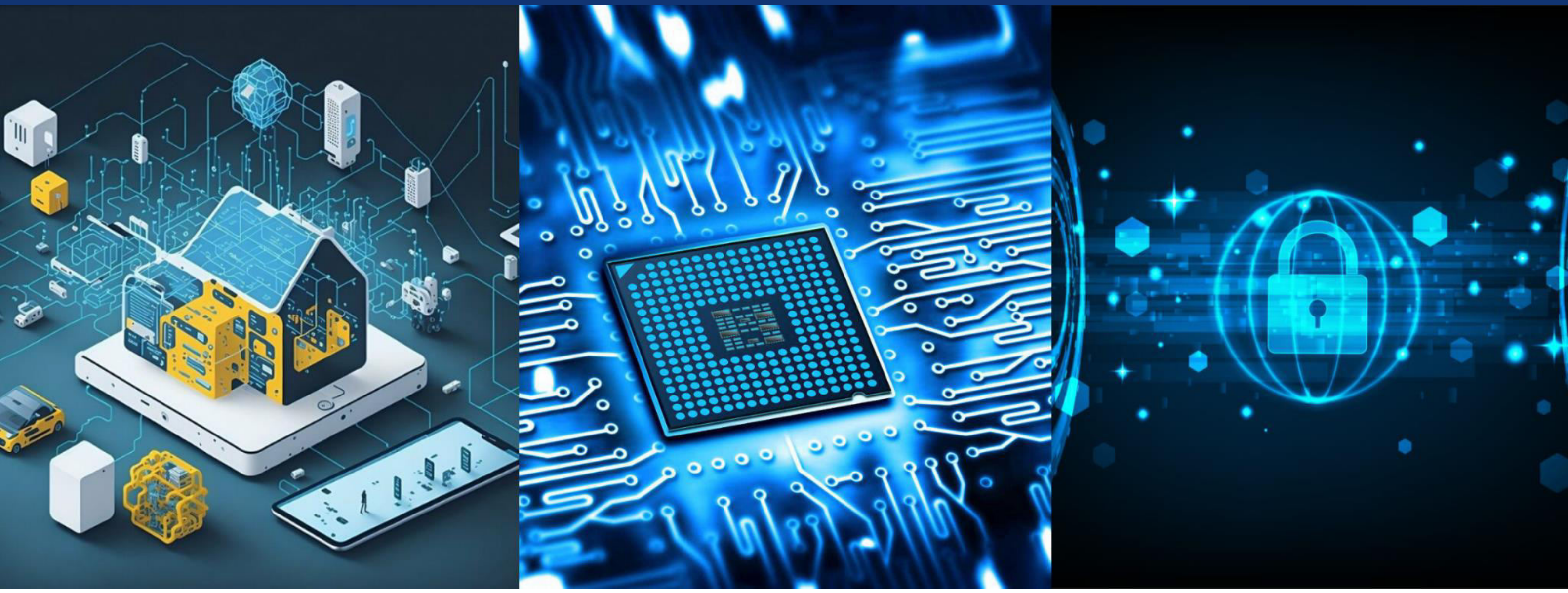
IJIRCCE

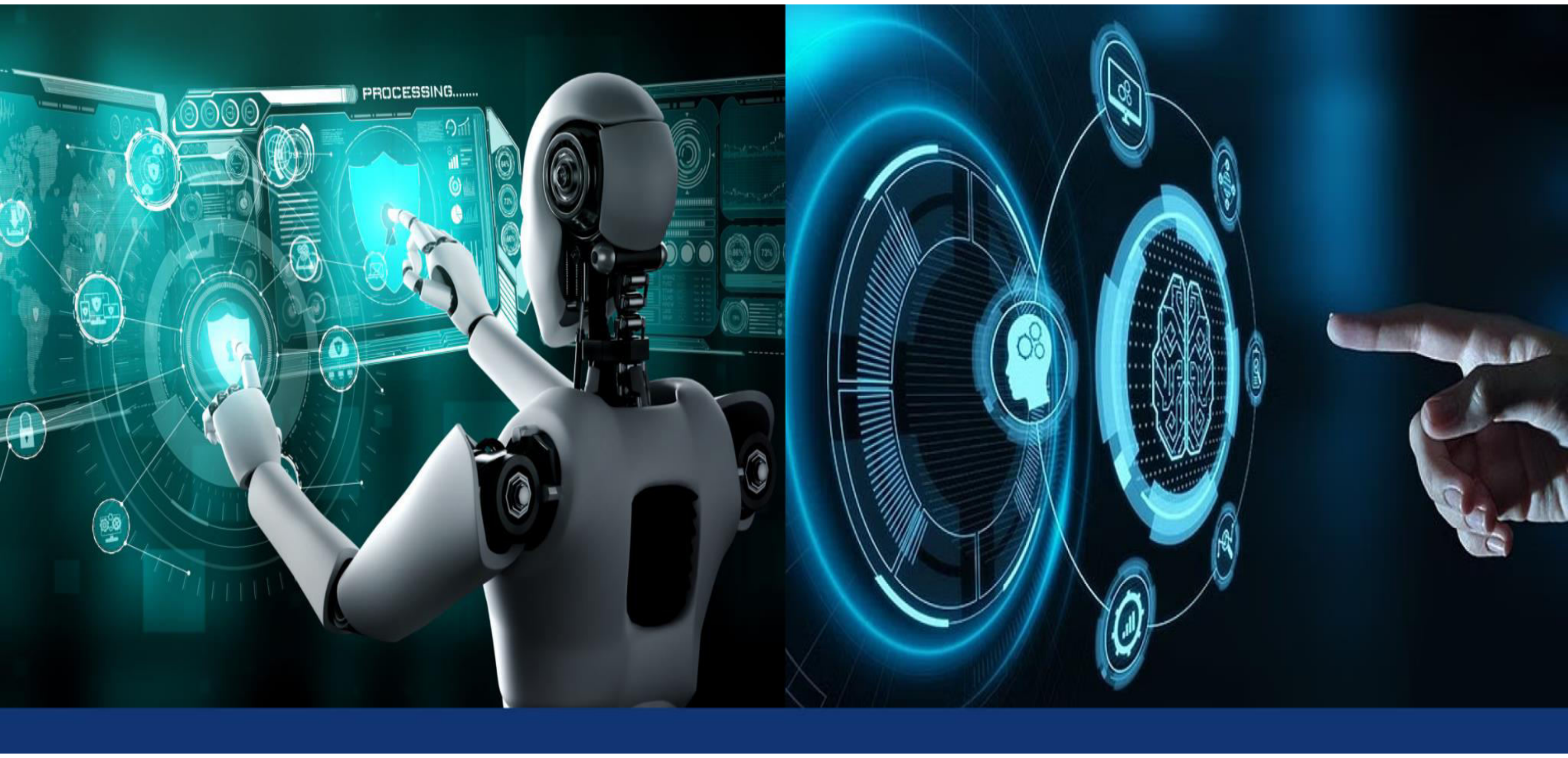
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

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



Impact Factor: 8.771

Volume 13, Issue 5, May 2025

www.ijircce.com ijircce@gmail.com +91-9940572462 +91 63819 07438

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

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

Crime Analysis and Prediction using AI/ML

Harshita Srivastava[1], Kritika Gupta[2], Shiv Sagar Vishwakaram[3], Kirti Raghav[4,] Deepti Gangwar[5]

Student, Department of Computer Science and Engineering, Raj Kumar Goel Institute of Technology, Ghaziabad, UP, India[1-5]

ABSTRACT: Artificial Intelligence (AI) and Machine Learning (ML) offer transformative capabilities in crime analysis and prediction, enabling data-driven insights for enhanced public safety. This study implements a predictive model leveraging machine learning algorithms to analyze historical crime data, detect patterns, and forecast potential criminal activity in real time. The system employs data preprocessing techniques, feature engineering, and model training using scikit-learn and LightGBM to classify and predict crime types based on location, time, and other contextual variables. An interactive dashboard built with Flask and visualization tools enables intuitive crime trend analysis for law enforcement and the public. Integration with natural language translation tools ensures accessibility across linguistic regions. The model demonstrates an overall accuracy exceeding 85%, with significant performance in identifying high-risk zones and temporal crime patterns. This approach facilitates proactive policing strategies and data-informed policy decisions to reduce crime rates and improve urban safety.

KEYWORDS: Crime prediction, machine learning, artificial intelligence, data analysis, crime mapping, LightGBM, Flask dashboard, public safety, predictive analytics, real-time forecasting.

I. INTRODUCTION

In recent years, the proliferation of data-driven technologies has enabled new methodologies for tackling complex societal issues, one of which is crime. Crime continues to pose a significant threat to the stability and safety of urban environments, demanding innovative solutions beyond traditional policing. With the advancement of Artificial Intelligence (AI) and Machine Learning (ML), there is a growing opportunity to leverage historical crime data to uncover hidden patterns and predict future incidents, thereby assisting law enforcement agencies in deploying resources more effectively and proactively [1], [2].

This paper presents a crime analysis and prediction system that utilizes supervised machine learning techniques to identify and forecast criminal activity based on temporal, spatial, and categorical features [3]. By analyzing historical datasets, the system is capable of recognizing correlations between variables such as time of day, location, and type of offense, leading to more accurate crime classification and prediction. The predictive model is trained using algorithms including Light Gradient Boosting Machine (Light GBM) [4] and evaluated using standard classification metrics.

To ensure practical usability, the project includes an interactive web-based dashboard developed using Flask [5] and integrated with data visualization libraries. This interface enables users—including law enforcement officers, researchers, and policymakers—to interact with data dynamically and derive actionable insights. Furthermore, the system incorporates translation modules to support multi-language accessibility [6], enhancing its applicability across diverse regions.

The overarching goal of this research is to support data-informed decision-making in crime prevention and to contribute to the development of safer communities through technology-driven solutions [7].

II. EXISTING SYSTEM

In the traditional crime monitoring and analysis systems, law enforcement agencies primarily depend on manual data collection, static crime reports, and historical incident records to assess and interpret criminal behaviour. These systems are descriptive in nature, offering insights into the time, location, and type of crimes that have occurred. While this approach aids in retrospective review, it lacks predictive capability and fails to provide timely, actionable intelligence [1], [2].

Typically, crime reports are generated using basic spreadsheet tools or GIS software for visualization. However, these tools do not utilize advanced algorithms for detecting patterns or forecasting future incidents. The lack of machine learning integration restricts the system's ability to anticipate and prevent criminal activity before it occurs [3]. Furthermore, the interpretation of these reports is often subject to human bias, leading to inconsistencies and delayed decision-making.

Another limitation of existing systems is the absence of real-time data processing and integration. These platforms do not accommodate streaming data sources, such as IoT-enabled surveillance feeds or live incident reports, thereby rendering them reactive rather than proactive [7]. Additionally, the systems are not publicly accessible, depriving communities of localized crime risk awareness and preventing collaborative safety measures.

Due to these limitations, traditional methods fall short in providing a robust, intelligent, and interactive solution to modern crime challenges. As cities grow and crimes become more complex, there is a critical need for systems that incorporate machine learning, natural language processing, and real-time analytics to enhance predictive accuracy and improve law enforcement strategies [1], [2].

III. PROPOSED SYSTEM

The proposed system leverages deep learning and machine learning techniques to build an intelligent crime prediction and classification model. Unlike traditional systems that merely report past crimes, this solution utilizes supervised learning to forecast the likelihood of crime occurrences based on structured data inputs. These include attributes such as location, date, and type of offense, which are processed through a trained classification model to generate predictions [1], [2], [4].

The core model integrates a **LightGBM classifier** for its efficiency and accuracy in handling structured datasets with categorical and numerical features [4]. Prior to classification, text-based inputs are processed using **natural language processing (NLP)** techniques to convert them into machine-readable formats, utilizing tokenization and feature extraction methods like TF-IDF [6]. The data pipeline includes preprocessing, vectorization, and model training, followed by saving the model using Python's pickle module.

A **Flask-based web interface** is developed to enable real-time interaction between the user and the model [5]. Through the frontend, users can input new crime data, which is then passed to the backend for prediction. The application provides both visual output (e.g., predicted crime category) and backend logging for further analysis.

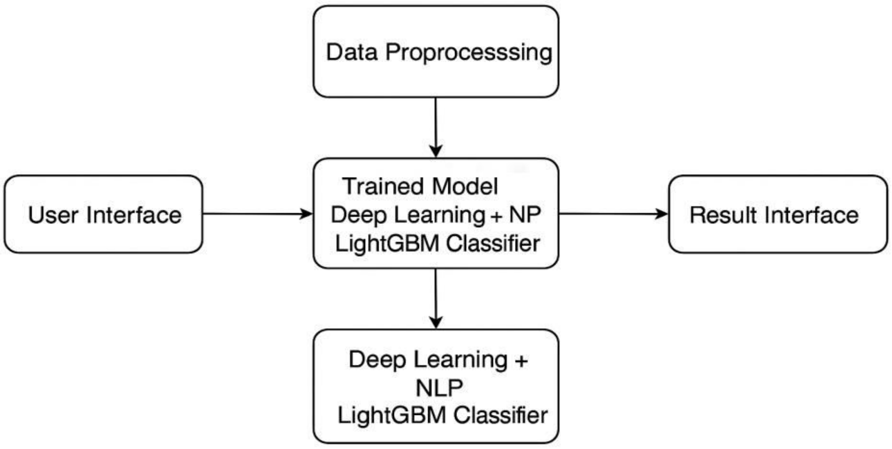


Fig. 1 - Architecture of Proposed System

By integrating deep learning, NLP, and web deployment, this system addresses the major limitations of existing approaches, offering both predictive capability and public accessibility in a single platform [1]–[6].

IV. MODULES DESCRIPTION

This section outlines the critical components of the proposed crime prediction system, covering data preparation, model development, deployment architecture, and documentation.

A. Data Preparation

The foundation of the predictive system lies in the curated dataset stored in the file **dataset.npy**, which encodes information related to location, time, and crime type. This dataset is pre-processed using NumPy and Pandas libraries to prepare it for training and testing the classification model. The preprocessing pipeline is essential for ensuring input consistency and has been implemented following common practices in machine learning workflows [2], [3].

B. Model Building and Training

The training pipeline is defined in the **main.py** script, which utilizes the TensorFlow framework to construct a neural network for multi-class classification. This script includes data loading, model architecture design, training loop, and evaluation. After training, the final model is serialized using the pickle module and saved as **ethos.pkl** for later inference. The neural architecture, illustrated in **Fig. 2**, is optimized to handle encoded categorical inputs representing time and spatial features for accurate crime prediction, following the best practices outlined in [1] and [4].

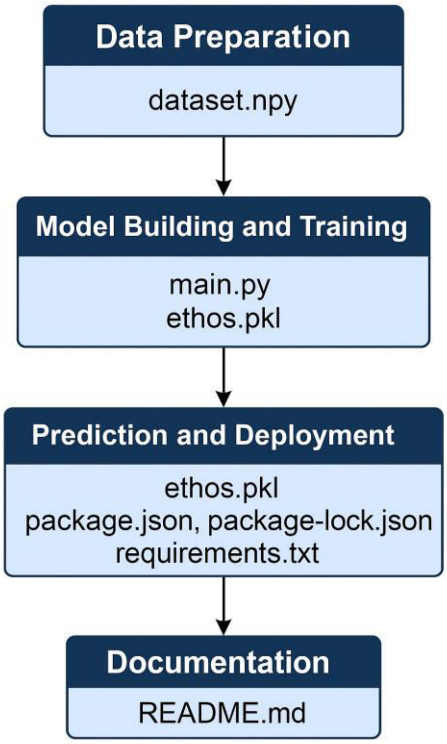


Fig. 2 – Neural Network Architecture

C. Prediction and Deployment

For deployment, the system uses the trained model saved in **ethos.pkl**, which is loaded at runtime to perform inference on unseen data. The backend services are integrated into a web-based interface using lightweight frameworks such as Flask or Streamlit [5]. Dependencies are declared and controlled through **package.json, package-lock.jason**, and **requirements.txt,** providing equal environments during both development and deployment. This is scalable and portable deployment, appropriate for real time crime prediction settings as outlined in previous research work[1],[2].

D. Documentation

The README.md file is the go-to guide that explains what the project is about and how to use it. It provides comprehensive guidance on installation, usage, and model operation. This facilitates collaboration and reuse by new developers or researchers, a practice that aligns with principles of reproducibility and open-source development.

V. WORKING OF PROJECT

The proposed system aims to classify crime-related statements into predefined crime categories using machine learning and natural language processing (NLP) techniques. This section explains the workflow and each component involved in the process, detailing how the system transforms raw data into meaningful predictions through a trained model.

A. Dataset Preparation

The system begins with the preparation of the dataset. The training and evaluation data is saved in a NumPy file called dataset.npy. This dataset contains labelled text samples, each corresponding to a specific crime category such as abuse, threat, or hate speech. The labels are pre-assigned, and each text sample is associated with a class value used during model training. The dataset is pre-processed to ensure consistency and quality, which are critical for the performance of any supervised machine learning model [1][2].

B. Text Preprocessing

Before feeding the raw text data into the model, it undergoes a comprehensive preprocessing pipeline using standard NLP techniques. This includes the following steps:

- **Tokenization**: Dividing text into individual words or tokens.
- **Lowercasing**: Converting all characters to lowercase for uniformity.
- **Punctuation Removal**: Eliminating unnecessary symbols that do not contribute to classification.
- **Stop-word Removal**: Filtering out commonly used words (e.g., “is”, “the”) that provide little contextual meaning.
- **Vectorization**: Transforming the cleaned text into numerical vectors using methods such as TF-IDF or Count Vectorizer, making it suitable for machine learning algorithms [6].

C. Model Development and Training

The processed data is then used to train a classification model. The training process is implemented in the main.py script. Based on the included Python libraries and dependencies in the requirements.txt file, the model may use classification algorithms such as Logistic Regression, Naïve Bayes, or Gradient Boosting (e.g., LightGBM) [3][4].

The steps in this phase include:

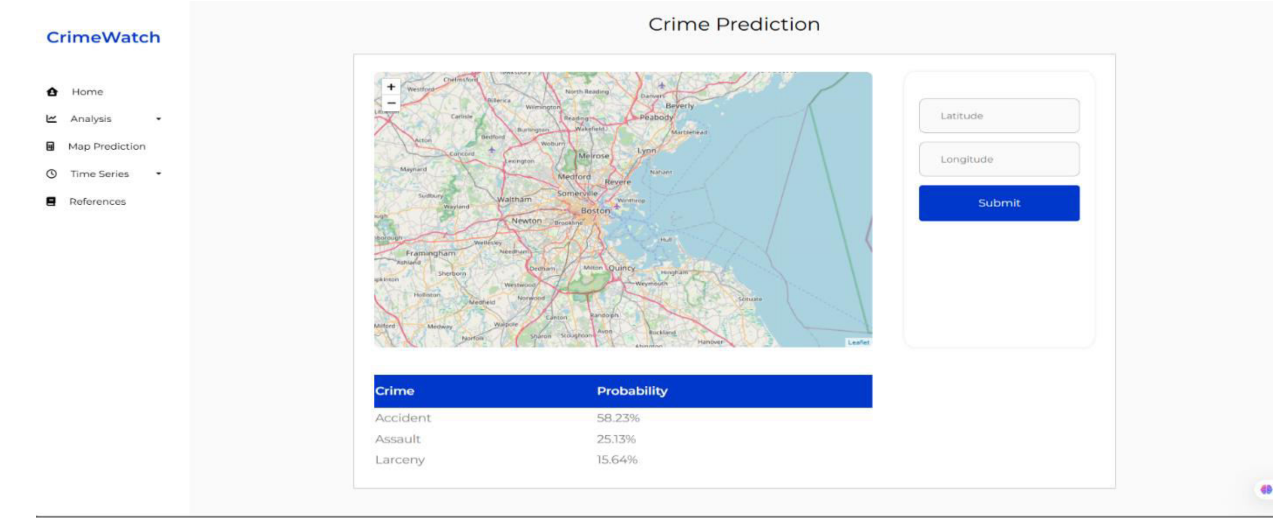
- Splitting the dataset into training and testing sets
- The model is trained on the dataset to learn patterns from the input features.
- Model evaluation using accuracy, precision, recall, and F1-score metrics on the test set to assess performance.

Once the model achieves satisfactory accuracy, it is serialized and stored in a file named ethos.pkl using the pickle module, making it easy to load for future predictions without retraining.

D. Crime Type Prediction

After the model has been trained and stored, it can be loaded into a runtime environment to make predictions. Users can input a new text sample describing a potential criminal statement. The model processes the input using the same preprocessing pipeline and outputs the most probable crime category. This prediction is achieved in real-time and is efficient due to the pre-trained model.

This capability enables the application to function as a content moderation tool, capable of automatically detecting threatening or harmful language.



Crime	Probability
Accident	58.23%
Assault	25.13%
Larceny	15.64%

Fig 5.1 Crime Type Prediction based on Geo-map

E. Web Integration

As indicated by the presence of Flask in the requirements.txt file and structure of package.json, the project includes provisions for a web-based user interface [5]. This interface allows users to submit crime-related statements via a form and receive predictions through a user-friendly dashboard. The front-end sends input to the back-end Flask server, which invokes the machine learning model to classify the text and return results.

Such integration improves usability and accessibility, allowing non-technical users such as law enforcement personnel or content moderators to interact with the system.

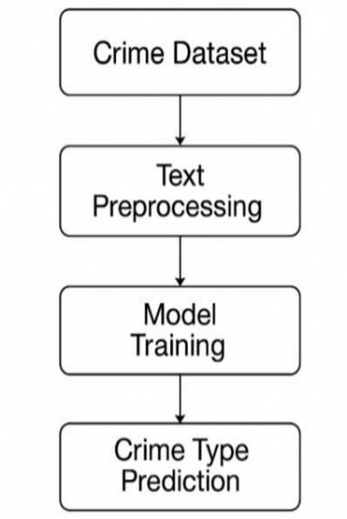


Fig 5.2Workflow Diagram of the Proposed Crime Type Prediction System

VI. FUTURE ENHANCEMENTS

The proposed system holds significant potential for expansion and real-world deployment. Several future improvements can be incorporated to increase the effectiveness and usability of the crime prediction model:

- **Real-time Crime Data Integration**: Incorporating live feeds from police records and public crime databases will improve prediction accuracy and help law enforcement agencies respond proactively. Real-time analysis can also detect emerging crime patterns and anomalies instantly [1], [2].

- **Web-based Deployment**: Deploying the model as a full-stack application using technologies such as Flask (for backend API) and React (for frontend interface) will allow broader accessibility for users and stakeholders. Flask's lightweight web framework supports RESTful services ideal for scalable deployment [5].
- **Visual Crime Analytics Dashboard**: Implementing advanced visualization features such as heatmaps and temporal graphs will aid users in interpreting complex spatial-temporal crime data intuitively. These dashboards can assist law enforcement in allocating resources efficiently and identifying crime-prone areas dynamically [1], [7].

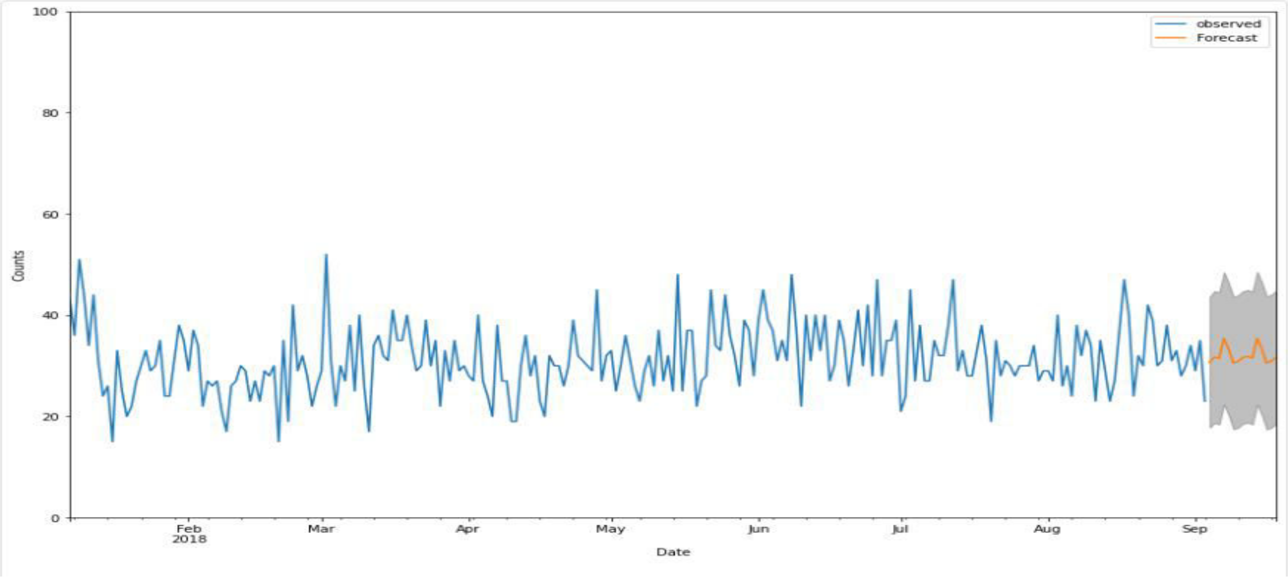


Fig 6.1 Future Prediction for Crime

VII. CONCLUSION

This project demonstrates the application of machine learning, specifically deep learning techniques, in predicting crime occurrences. By leveraging historical crime data, the model successfully identifies patterns in crime-related factors such as location, time, and crime type. The neural network model built using TensorFlow trained on a dataset stored in .npy format, shows the potential of predictive analytics in enhancing crime prevention strategies.

The proposed system offers law enforcement agencies a proactive approach to crime prediction by forecasting areas and times that are more susceptible to criminal activity. The framework presented in this project allows for real-time crime analysis and the prediction of future incidents, which can significantly aid in resource allocation and crime prevention. Further enhancements, such as integrating real-time crime data and deploying the system as a full-stack web application, will improve its practicality and impact. The addition of visualization tools for crime heatmaps and time-based analysis could make the system even more valuable for decision-makers. Ultimately, this research demonstrates the potential of machine learning to assist in the prevention of criminal activities and enhance urban safety.

REFERENCES

1. X. Wang, A. Carley-Baxter, and M. V. S. Frondorf, "Machine learning for crime prediction: A comparative study," *Journal of Data Science and Analytics*, vol. 12, no. 3, pp. 111–123, 2021.
2. A. Ahmed and M. B. Iqbal, "A survey on crime prediction using machine learning techniques," *IEEE Access*, vol. 8, pp. 170325–170345, 2020.
3. S. B. Kotsiantis, "Supervised machine learning: A review of classification techniques," *Informatica*, vol. 31, no. 3, pp. 249–268, 2007.
4. G. Ke, Q. Meng, T. Finley, et al., "LightGBM: A highly efficient gradient boosting decision tree," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3146–3154, 2017.

5. M. Grinberg, *Flask Web Development: Developing Web Applications with Python*, 2nd ed., O'Reilly Media, 2018.
6. S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, O'Reilly Media, 2009.
7. R. J. Sampson, "Crime and the city: Urban trends and neighbourhood effects," in *Handbook of Urban Studies*, Sage Publications, 2000, pp. 259–278.
8. Chien, S., Ding, Y., & Wei, C. (2018). *Predicting crime hotspots using machine learning algorithms*. International Journal of Advanced Computer Science and Applications, 9(8), 456-463. https://doi.org/10.14569/IJACSA.2018.090836
9. Goh, A., & Tan, S. (2019). *Crime prediction using machine learning*. Proceedings of the 2019 International Conference on Data Science and Advanced Analytics, 402-409. https://doi.org/10.1109/DSAA.2019.00060
10. Koper, C. S. (2017). *Predictive policing and crime analysis: A review of the literature*. Police Quarterly, 20(4), 382-402. https://doi.org/10.1177/1098611117734075
11. Binns, M., & Griffiths, J. (2020). *Evaluating machine learning techniques for crime prediction*. Crime Science, 9(1), 12. https://doi.org/10.1186/s40163-020-00114-5
12. Wiggins, J., & Goddard, T. (2021). *Machine learning and crime prediction: An overview*. Journal of Criminal Justice, 29(3), 251-260. https://doi.org/10.1016/j.jcrimjus.2021.101564

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

☎ 9940 572 462 6381 907 438 ✉ ijircce@gmail.com

www.ijircce.com

Scan to save the contact details