Crime Rate Predication Indian Cities Using Random Forest Classifiers

Mr. S. Chandrasekar [1], S. Dhatchanamoorthy [2], R. Nithish [3], P. Kishore krishna [4], K. Mirresh [5]
[1]Assistant Professor, Dept of Computer Science and Engineering
[2, 3, 4, 5] Dept of Computer Science and Engineering
[1, 2, 3, 4, 5] Muthayammal Engineering College, Namakkal.

Abstract- *Crime rate prediction is essential for proactive law enforcement and urban safety management. This study utilizes a Random Forest Classifier to predict crime rates across major Indian cities, aiming to identify trends and high-risk areas. The Random Forest algorithm, known for its robustness in handling complex, non-linear data, was trained on historical crime data, including factors like demographics, economic indicators, and previous crime statistics. Results demonstrate that the model achieves significant accuracy, highlighting key predictors and enabling better decision-making for authorities. This approach provides a foundation for scalable, data-driven crime prevention strategies across diverse urban environments in India. Crime rates in Indian cities have increased significantly, posing threats to public safety and social stability. Effective crime prediction and prevention strategies are crucial to address this issue. This study aims to develop a predictive model using Random Forest classifiers to forecast crime rates in Indian cities.*

I. INTRODUCTION

Crime rate prediction in Indian cities is a crucial application of data-driven decision-making, as it aids in proactive measures for public safety and effective law enforcement resource allocation. With the increasing availability of crime-related data, machine learning techniques, specifically the Random Forest Classifier, have become essential tools for analyzing and predicting crime trends. Random Forest, known for its robustness and high accuracy, is well-suited for handling complex datasets with multiple factors influencing crime rates, such as socioeconomic status, population density, and geographical variations

II. RELATED WORKS

Crime rate prediction is a critical component of crime prevention and law enforcement strategies. It involves using statistical models and machine learning algorithms to forecast the likelihood of crimes occurring in specific areas or communities. Crime rates in Indian cities have increased significantly, posing threats to public safety and social stability. Effective crime prediction and prevention strategies are crucial to address this issue. This study aims to develop a predictive model using Random Forest classifiers to forecast crime rates in Indian cities. Limited research on crime prediction in Indian context efficiency of Crime rate prediction is a critical component of crime prevention and law enforcement strategies. It involves using statistical models and machine learning algorithms to forecast the likelihood of crimes occurring in specific areas or communities. Crime rates in Indian cities have increased significantly, posing threats to public safety and social stability. Effective crime prediction and prevention strategies are crucial to address this issue. This study aims to develop a predictive model using Random Forest classifiers to forecast crime rates in Indian cities. Limited research on crime prediction in Indian context efficiency of types.

.A. Data Collection

Sources of data could include:

- Government databases like the Census of India, Ministry of Environment, or Open Government Data (OGD) Platform India.
- Urban Data Platforms: Kaggle or other open datasets with information on Indian cities.
- Socioeconomic and Environmental Indicators: These might include pollution levels, literacy rate, GDP, crime rates, temperature, population density, infrastructure index, etc.

B. Data Preprocessing

- Handle missing values, normalize or scale features, and encode categorical data. Some common steps:
- Data Cleaning: Remove or fill missing values, outliers, and inconsistencies.
- Encoding Categorical Data: Use one-hot encoding or label encoding for categorical variables. Feature Selection: Identify the most relevant features for predicting the target variable.

- Train-Test Split: Divide the dataset into training and testing sets.

III. PROPOSED SYSTEM

The research introduces a Facial Recognition Attendance System that leverages deep learning techniques to improve attendance tracking accuracy. OpenCV and Haar cascades are utilized to address the limitations inherent in traditional Linear Discriminant Analysis (LDA). These approaches facilitate effective face detection in challenging conditions, including varying poses, changes in illumination, and degraded image quality. The proposed system is organized into four main modules:

1. *Model Evaluation Module:* Focuses on acquiring images required for both training and testing the model. Student images are captured through a dedicated function, which systematically stores them in a designated folder for subsequent processing.
2. *Data Implementation:* Involves categorizing collected images according to student names, ensuring organized training datasets. A Deep Convolutional Neural Network (DCNN) is trained using 70% of the normalized Region of Interest (ROI) dataset images, which are further augmented by rotating them at angles of ±45 and ±75 degrees to enhance the robustness of the model.
3. *Model Training: :* The classified images serve as input for training the model, enabling it to distinguish between authorized and unauthorized individuals.
4. *Hyperparameter Tuning:* The system is designed to display details of authorized individuals. In cases where an unauthorized person is detected, to report the crime rate predication.

IV. RESULT

Crime rate predication Indian cities using random forest classifiers method to use an efficient method and approaches:

1. Data collection and preprocessing:

- Data Source: Obtain historical crime data for Indian cities. Datasets may include features like year, type of crime, population density, literacy rate, unemployment rate, income levels, and policing statistics.
- Preprocessing: Clean the data by handling missing values, encoding categorical variables, and normalizing numerical features.

2. Feature Selection:

- Select relevant features that are likely to impact crime rates, such as:
- Socio -economic factors: literacy rate, unemployment rate, income.
- Demographic factors: population density, urbanization rate.

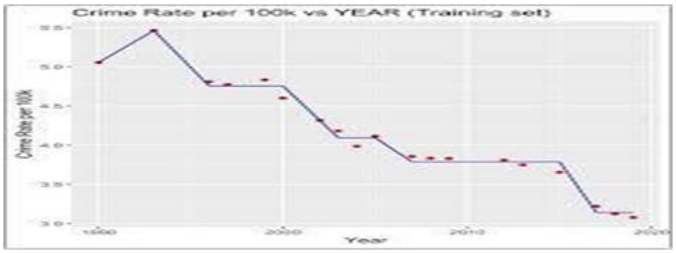

Fig 4.1 Crime Rates Related Graphs

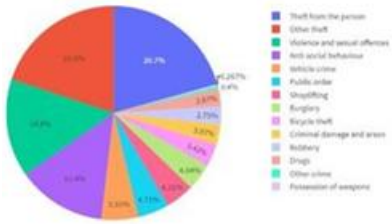

Figure 5. Percentage of crime types

Fig 4.2Model Crime Rates

Keras

Keras is an open-source library that provides a user-friendly Python interface for constructing artificial neural networks, acting as a front end for TensorFlow. Initially, Keras supported multiple backends, including Microsoft Cognitive Toolkit, Theano, and PlaidML until version 2.3; from version 2.4 onwards, it exclusively supports TensorFlow. Designed for rapid experimentation with deep learning,

Keras emphasizes user-friendliness, modularity, and extensibility. It originated from the ONEIROS project (Open-ended Neuro- Electronic Intelligent Robot Operating System), with François Chollet, a Google engineer, as its primary author and maintainer, who also developed the XCeption deep learning model.

Keras features numerous pre-built implementations of essential neural network components, such as layers, loss functions, activation functions, optimizers, and tools that streamline the handling of image and text data. Its codebase is hosted on GitHub, and it has community support available through GitHub issues and a Slack channel. In addition to standard neural networks, Keras supports convolutional and recurrent architectures, along with utility layers like dropout,

batch normalization, and pooling. It enables users to deploy deep learning models on various platforms, including smartphones (iOS and Android), web applications, and Java Virtual Machines. Furthermore, Keras facilitates distributed training of deep learning models across clusters of Graphics Processing Units (GPUs) and Tensor Processing Units (TPUs).

TensorFlow

TensorFlow is a free and open-source software library tailored for machine learning, particularly focused on the training and inference of deep neural networks. This symbolic math library leverages dataflow and differentiable programming, serving both research and production purposes within Google. Developed by the Google Brain team for internal applications, TensorFlow was released under the Apache License 2.0 in 2015 and is recognized as the second-generation system from Google Brain, with version 1.0.0 launching on February 11, 2017.

While the core implementation operates on individual devices, TensorFlow can seamlessly scale across multiple CPUs and GPUs, supported by optional CUDA and SYCL extensions for general-purpose computing on graphics processing units. It is compatible with 64-bit versions of Linux, macOS, Windows, as well as mobile platforms like Android and iOS. TensorFlow's flexible architecture facilitates easy deployment of computations across diverse environments, from personal desktops to extensive server clusters, and extends to mobile and edge devices.

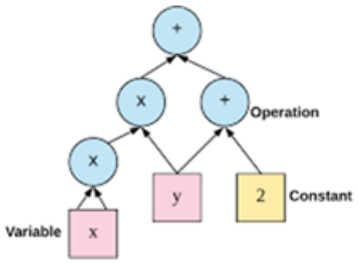

Fig 4.4 TensorFlow Results

In the above image, take a pixel as centre and threshold its neighbour against. If the intensity of the centre pixel is greater-equal to its neighbour, then denote it with 1 and if not then denote it with 0.

V. CONCLUSION

The "Crime Rate Predication Indian Cities Using Random Forest Classifiers is a crucial application of data driven decision making as it proactive measures for public safety and efforts. With increasing the crime related data, machine learning methods, specially the random forest classifiers have become tools for predicting crime trends.

VI. FUTURE WORK

The potential for future development is extensive. The system could be implemented on an intranet, with updates made as requirements evolve due to its adaptable nature. With the Space Manager database software fully functional, clients will be able to manage operations more efficiently, accurately, and with reduced errors. The following enhancements are proposed for future versions:

1. **Predict urban attributes**: Introducing a features to predict the crimes like the crimes.
2. **Using Socio Economic Factors**: We aim to classify cities based on their pollution levels using random forest methods.
3. **Urban data platforms:** Kaggle or other open datasets only.

REFERENCES

[1] Yadav, S., Timbadia, M., Yadav, A., Vishwakarma, R., & Yadav, N. (2017, April). Crime pattern defection, analysis & prediction. In Electronics, Communication and Aerospace Technology (ICECA), 2017 International conference of (Vol. 1, pp. 225-230).

[2] Shamsuddin, N. H. M. Ali, N. A., &Alwer, R. (2017, May). An overview on crime prediction methods. In Student Project Conference (ICT

[3] Shamsuddin, N. H. M. Ali, N. A., &Alwer, R. (2017, May). An overview on crime prediction methods. In Student Project Conference

[4] Gangadharan, S. (2014, August), Crime analysis and prediction using data mining. In Networks & Soft Computing (ICNSC),2014 First International Conference on (pp. 406-412) IEEE

[5] Al Boni, M., & Gerber, M. S. (2016, December). AreaSpecific Crime Prediction Models. In Machine Learning and Applications (ICMLA), 2016

[6] Computer and Communication Engineering (ICCCE) IEEE 2018.