Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

A Novel Fusion Approach for Advancement in Crime Prediction and Forecasting using Hybridization of ARIMA and Recurrent Neural Networks

Thayyaba Khatoon Mohammed[1*], D N Vasundhara[2], Syeda Husna Mehanoor[3], E. Sreedevi[4], Puranam Revanth Kumar[5], CH Manihass[6], Shaik Fareed Baba[7]

[1,2]Department of Computer Science and Engineering, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad, India.
[3]Department of Computer Science and Engineering, Malla Reddy College of Engineering for Women, Hyderabad, India.
[4]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh
[5, 6, 7] Department of Artificial Intelligence and Machine Learning, Malla Reddy University, Hyderabad, India.
[*]Email: thayyabakhatoonmohammed@gmail.com

ARTICLE INFO

Received: 30 Dec 2024

Revised: 19 Feb 2025

Accepted: 27 Feb 2025

ABSTRACT

Introduction: Law enforcement agencies have struggled to forecast crime due to the complexity of criminal conduct and its dynamic elements. Regular developments and the growth of well-known and new platforms have made social media a pervasive way for people to express their ideas and experiences, popularizing blogging and democratizing information distribution. Traditional time series models like Autoregressive Integrated Moving Average (ARIMA) are frequently employed for forecasting, but they may struggle to capture crime data's nonlinear relationships and long-term dependencies. However, Long Short-Term Memory (LSTM) are best at catching sequential patterns but may struggle with short-term data or rapid criminal patterns. Combining strength of both methods may improve forecast accuracy and robustness.

Objectives: To develop a hybrid ARIMA-LSTM model that effectively captures both linear temporal trends and complex nonlinear patterns in crime data, enhancing predictive accuracy across diverse crime types and geographic regions in India.

Methods: The proposed fusion approach leverages the strengths of ARIMA in modelling temporal dependencies and the ability of LSTMs to capture complex nonlinear relationships. Initially, ARIMA is employed to model the underlying trend and seasonality in the crime data. Subsequently, the residuals obtained from ARIMA are fed into an LSTM architecture, such as LSTM, to capture the remaining nonlinear patterns and dependencies. The hybrid model is trained using historical crime data and validated using appropriate evaluation metrics. The performance of the proposed fusion approach is evaluated on real-world crime datasets across various geographic locations and crime types in India.

Results: Experimental results demonstrate that the hybrid ARIMA-LSTM model outperforms individual methodologies and baseline models in terms of MSE, RMSE, and accuracy.

Conclusions: This research contributes to the development of more effective crime prediction models, aiding law enforcement agencies in proactive decision-making and resource allocation for crime prevention.

Keywords: ARIMA, Crime prediction, Deep learning, LSTM, Forecasting, Recurrent neural network.

INTRODUCTION

In recent years, the growth of data analytics and machine learning has set up new paths for tackling difficult social problems, including crime prediction. Researchers and practitioners in the field of law enforcement have begun to employ novel approaches in order to anticipate and avoid illegal acts, acknowledging the significance of proactive strategies [1]. Crime is a major societal issue now. Greater urban areas have a more significant crime rate than less crowded areas. The rising crime rate in cities is a major issue in a lot of nations. Crime evaluation procedures are necessary to decrease crime rates, which are on the rise [2]. Allocating patrol officers based on the crime rate is one way to decrease criminal activity. It is challenging, though, to make effective predictions about future criminal activity. There are a few various types of crimes that can be classified, such as violent crimes and nonviolent crimes categories [3]. The criminal's use of physical force or threats against a specific victim constitutes a violent crime. The severity of these offenses is determined to be higher than that of nonviolent offenses [4]. Crimes including murder, aggravated assault, abduction, robbery, and forced rape are all part of a violent act [5, 6]. The use of the weapon in a violent crime is debatable. The recording and reporting of crimes also vary across countries. Analysis of the ever-increasing amount of criminal data in an efficient and accurate manner is the principal challenge that law enforcement agencies are currently facing. The problem is that many security forces don't have what it takes to tackle this problem; they don't have the training or resources to effectively sort through all this data. When it comes to obtaining the subtle insights needed to effectively combat crime, traditional techniques of analysis, such manual inspection of incident reports or basic statistical analysis, frequently fall short [7]. Figure 1 describes flow of crime prediction

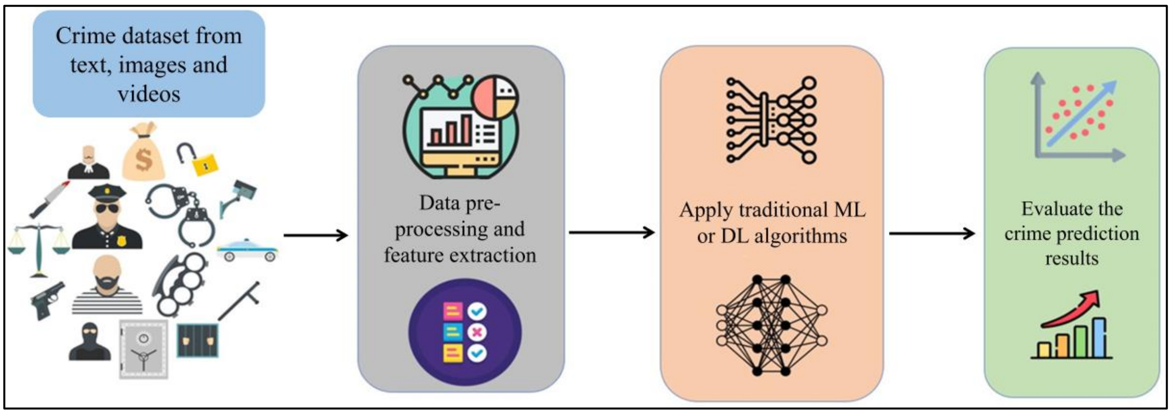

Crime dataset from text, images and videos
Data pre-processing and feature extraction
Apply traditional ML or DL algorithms
Evaluate the crime prediction results

Figure 1. Represents the flow of crime prediction

Criminal conduct is inherently complex due to its inherent fluidity, the offenders' ever-changing methods, and the complex web of influences on criminal activity. The necessity for cutting-edge analytical tools and procedures is further highlighted by the fact that traditional methods become irrelevant in the face of such rapid change. In addition to being able to process massive amounts of data, these cutting-edge instruments should be able to decipher its underlying complexities [8]. The availability of such tools could lead to the discovery of hidden trends and patterns, providing law enforcement with the data they need to make educated judgments and allocate resources efficiently. Ultimately, the continuous fight against crime hinges on the pursuit of ever-more-advanced analytical capacities. The use of data analytics and machine learning by law enforcement organizations can yield priceless insights, allowing for better public safety and security through proactive and targeted interventions [9]. Law enforcement professionals and academics have been working tirelessly to develop new approaches to proactive strategy development with the goal of detecting and preventing illegal acts. Recent advances in deep learning, especially the use of LSTM, have emerged as powerful tools for improving crime prediction models; these innovations are among the most promising in this field [10].

Table 1. List of abbreviations

Term	Abbreviation
Autocorrelation Function	ACF
Artificial Neural Network	ANN
Auto-Regressive	AR
Autoregressive integrated moving average	ARIMA
Convolutional Neural Network	CNN
Comma Separated Values	CSV

Gated Recurrent Unit	GRU
Long Short-Term Memory	LSTM
Moving Average	MA
Neural Networks	NNs
Partial Autocorrelation Function	PACF
Root Mean Squared Error	RMSE
Recurrent Neural Network	LSTM

The analytical capacities of law enforcement agencies could be greatly improved by utilizing deep learning techniques, especially when it comes to analysing crime data. LSTMs stand out because of their capacity to grasp the temporal relationships included in sequential data. This allows for a more sophisticated comprehension of behavioural patterns and makes it easier to spot small changes and new trends in criminal actions. While LSTMs excel at catching short-term dynamics, they could struggle to spot the underlying seasonal patterns and long-term trends in crime data. The necessity for a more comprehensive strategy capable of handling both short-term variations and long-term patterns is highlighted by this constraint [11]. Consequently, there has been a lot of focus on creating a new hybrid method that combines ARIMA and LSTMs. This novel technique seeks to offer a thorough framework for crime prediction by combining the temporal modelling skills of LSTMs with the time series forecasting capabilities of ARIMA. With the hybrid model, we may take advantage of the best features of both approaches while avoiding their drawbacks [12]. The hybrid technique shows promise for improving the accuracy and resilience of crime prediction models by combining ARIMA's skill in capturing long-term trends and seasonal patterns with LSTMs' skill in catching short-term dynamics. Our methodology also aims to help strengthen public safety efforts by advancing crime prediction and fusing varied approaches with contextual information. By combining ARIMA and LSTMs into one cohesive framework, law enforcement organizations may better understand criminal behaviour, which in turn allows them to proactively fight crime and protect communities [13].

The rest of the paper is organized as follows: Section 2 presents the related work. Section 3 describe the methodology. Section 4 presents the experimental evaluation of the proposed technique. The outcomes are concluded in Section 5.

LITERATURE SURVEY

Globally, law enforcement organizations and researchers have been interested in the topic of crime prediction for many years. For a long time, anticipating criminal actions has relied on traditional methodologies, such as statistical time series models like ARIMA [14]. But the complicated dynamics in crime data are often too much for these approaches to handle. Because of their capacity to grasp sequential patterns and nonlinear correlations, Recurrent Neural Networks (LSTMs) have been investigated by [15] as a potential tool for crime prediction in light of recent developments in AI and ML. Spatial-temporal crime prediction challenges, was proposed by [16] which take into account both the geographical distribution and temporal patterns of crime scenes, deep learning models like CNNs and LSTMs have proven to be rather effective. The models achieve better prediction accuracy by incorporating both spatial and temporal information [17]. They use convolutional layers to extract spatial characteristics and recurrent layers to capture temporal connections was proposed by [18]. Another application of deep learning models introduced by [19] is anomaly detection in crime data, which seeks to spot out-of-the-ordinary patterns that can point to new dangers or suspicious behaviour. The use of deep learning methods has also improved multimodal crime analysis by [18]. These methods allow for the combination of structured crime data with unstructured text data gathered from social media and multimedia footage captured by surveillance cameras. By integrating data from various sources, deep learning models [20, 21] improve situational awareness and forecast accuracy by providing a more holistic view of criminal dynamics. With the use of transfer learning and domain adaptation approaches, models trained on similar domains or datasets can be adjusted to fit new, distinct contexts, thus expanding the use of deep learning for crime prediction [22].

Another step toward making crime prediction systems more transparent and interpretable is the rise of explainable deep learning models. Attention mechanisms [23] and saliency maps [24] are two techniques that help stakeholders understand the elements impacting predictions and develop trust in predictive models. These tools highlight crucial aspects and decision-making processes. When it comes to detecting nonlinear patterns and temporal connections in sequential data, the Gated Recurrent Unit (GRU) [25] has demonstrated encouraging results. The author of [26] used LSTM networks to forecast the frequency of crimes in cities, and their predictions were more accurate than those made using more conventional methods.

The ability of NNs, and specifically NN designs such as LSTM and GRU [27], to detect nonlinear patterns and temporal connections in sequential data is encouraging. When used separately for crime prediction tasks, ARIMA [28] and LSTM [29] both have significant limits, despite their particular benefits. When dealing with short-term data or unexpected changes in crime trends, LSTMs may not perform well, and ARIMA models may have trouble capturing complicated nonlinear correlations. Researchers [30, 31] have suggested hybrid methods that merge the best features of both approaches to overcome these constraints. The performance of applying ARIMA and LSTMs separately to crime prediction problems has been mixed, but combining the two approaches shows promise for improving prediction accuracy and robustness. The complex dynamics of crime data can be better captured by hybrid approaches, which combine the capabilities of both methodologies. This, in turn, helps law enforcement agencies with proactive decision-making and resource allocation for crime prevention and control [32].

METHODS

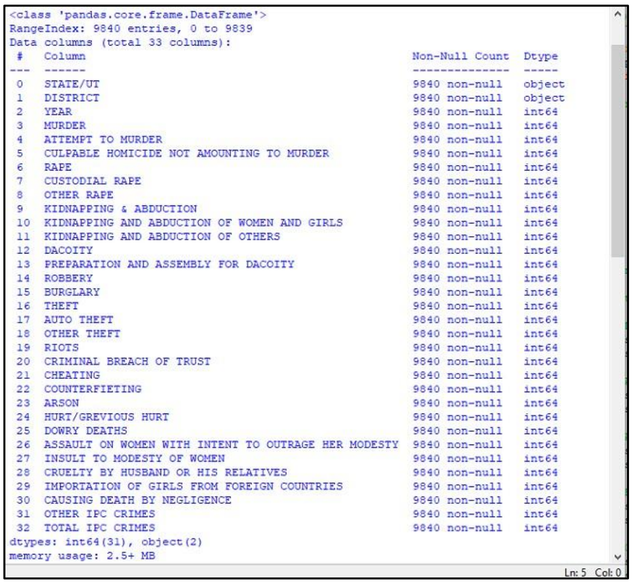
There are both linear and nonlinear components in the well production data, which are standard times series data. Previous research confirms that ARIMA, a time-tested linear statistical approach, is an effective tool for time series forecasting. On the other side, LSTM is able to acquire datasets that contain nonlinear characteristics. We propose ARIMA-LSTM model that integrate both linear and nonlinear components. This will reveal the impact of the open-shut manual well operations, which cause nonlinear oscillations in production data.

Dataset Description

The data that were utilized in this study are real-time datasets that were obtained from the database that details criminal activity in India. A number of attributes are included in the dataset, including: FID, Record Id, Offense Code, Offense Extension, Offense Category, Description, Police District, Beat, Grid, and occurrence Timestamp. A crime record complete with a timestamp and date is stored in each instance of the dataset. Figure 2 shows that there are a total of 32 distinct types of crimes. There are around 71,243 records in the data set, which is a compilation of information from the previous nine years.

Pre-Processing

During the data preprocessing phase, several systematic techniques are employed to ensure data quality and to prepare the dataset for effective analysis and model training. The process begins with addressing missing values, which are common in real-world datasets. For numerical features, the missing values are imputed using the mean of the corresponding column to maintain the statistical distribution of the data, whereas for categorical variables, missing entries are replaced with a placeholder value such as 'Unknown' to retain the categorical integrity without introducing noise. To eliminate redundancy and potential biases in the dataset, duplicate entries are removed using the df.drop_duplicates() method, ensuring that each data point is unique and meaningful. Time-related attributes are standardized by converting timestamp fields into datetime format using pd.to_datetime(), which facilitates the extraction of additional temporal features like year, month, day, hour, and day of the week through pandas' dt accessor functions. These derived temporal features are essential for capturing cyclical and seasonal trends in crime patterns. Numerical features are scaled using the MinMaxScaler() to normalize values within a defined range, which is crucial for ensuring uniformity across features and for optimizing the performance of gradient-based learning algorithms. Categorical variables, such as offense types or location identifiers, are encoded using a combination of LabelEncoder() for ordinal representation and pd.get_dummies() for one-hot encoding to convert them into a machine-readable format without implying any ordinal relationship where none exists. Class imbalance, a common issue in crime datasets where certain crime types are overrepresented compared to others, is addressed using the Synthetic Minority Over-sampling Technique (SMOTE) from the imblearn.over_sampling module, which generates synthetic examples for minority classes to achieve a balanced class distribution. In addition to these steps, feature engineering is conducted by creating new features derived from existing date and location-based information, helping to enhance the model’s contextual understanding. Correlation analysis and statistical significance tests are used for feature selection, ensuring that only the most relevant and influential features are retained for model training. Finally, the cleaned and transformed dataset is split into training and testing sets using the train_test_split() function with an 80-20 ratio, ensuring that the model is evaluated on unseen data and generalizes well to future predictions..



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9840 entries, 0 to 9839
Data columns (total 33 columns):
 #   Column                                               Non-Null Count  Dtype
---  ------                                               --------------  -----
 0   STATE/UT                                             9840 non-null   object
 1   DISTRICT                                             9840 non-null   object
 2   YEAR                                                 9840 non-null   int64
 3   MURDER                                               9840 non-null   int64
 4   ATTEMPT TO MURDER                                    9840 non-null   int64
 5   CULPABLE HOMICIDE NOT AMOUNTING TO MURDER            9840 non-null   int64
 6   RAPE                                                 9840 non-null   int64
 7   CUSTODIAL RAPE                                       9840 non-null   int64
 8   OTHER RAPE                                           9840 non-null   int64
 9   KIDNAPPING & ABDUCTION                               9840 non-null   int64
 10  KIDNAPPING AND ABDUCTION OF WOMEN AND GIRLS          9840 non-null   int64
 11  KIDNAPPING AND ABDUCTION OF OTHERS                   9840 non-null   int64
 12  DACOITY                                              9840 non-null   int64
 13  PREPARATION AND ASSEMBLY FOR DACOITY                 9840 non-null   int64
 14  ROBBERY                                              9840 non-null   int64
 15  BURGLARY                                             9840 non-null   int64
 16  THEFT                                                9840 non-null   int64
 17  AUTO THEFT                                           9840 non-null   int64
 18  OTHER THEFT                                          9840 non-null   int64
 19  RIOTS                                                9840 non-null   int64
 20  CRIMINAL BREACH OF TRUST                             9840 non-null   int64
 21  CHEATING                                             9840 non-null   int64
 22  COUNTERFIETING                                       9840 non-null   int64
 23  ARSON                                                9840 non-null   int64
 24  HURT/GREVIOUS HURT                                   9840 non-null   int64
 25  DOWRY DEATHS                                         9840 non-null   int64
 26  ASSAULT ON WOMEN WITH INTENT TO OUTRAGE HER MODESTY  9840 non-null   int64
 27  INSULT TO MODESTY OF WOMEN                           9840 non-null   int64
 28  CRUELTY BY HUSBAND OR HIS RELATIVES                  9840 non-null   int64
 29  IMPORTATION OF GIRLS FROM FOREIGN COUNTRIES          9840 non-null   int64
 30  CAUSING DEATH BY NEGLIGENCE                          9840 non-null   int64
 31  OTHER IPC CRIMES                                     9840 non-null   int64
 32  TOTAL IPC CRIMES                                     9840 non-null   int64
dtypes: int64(31), object(2)
memory usage: 2.5+ MB
```

Figure 2. Input data formation

This comprehensive preprocessing and feature engineering pipeline ensures that the data fed into machine learning models is clean, balanced, and rich in meaningful information, thereby enhancing model robustness and predictive performance.

Proposed ARIMA Model

ARIMA models were created for stationary time series. Adding a phase to remove non-stationarity in a time series created a new class of models. Box and Jenkins created stochastic ARIMA models. Time series identification, estimate, and verification comprise ARIMA's three-stage iterative model. The Box-Jenkins method relies on integration, AR, and MA filters. An integration filter observable data into a differenced series. The AR filter generates an intermediate series that the MA filter processes into random white noise.

The formula for an ARIMA (p, d, q) model is:

$$\left(\mathrm{ARIMA(p, d, q)} = \& \, (n) = \mu + \theta 1(n-1) + \cdots + \theta p(n-p) @\& \; + \theta(n) + \theta 1(n-1) + \cdots + \theta q(n-q)\right) \tag{1}$$

In equation (1), the variable μ represents the mean of the series, the coefficients $\theta 1, \dots, \theta p$ represent the autoregression coefficients, and the coefficients $\theta 1, \dots, \theta q$ represent the moving average coefficients, this is the value that is anticipated at time f(n). The current time step is denoted by $n-1, n-2, \dots, n-p$. The following are the three components that can be used to break down this equation: The autoregression (AR) is a statistical term that indicates the linear dependence that exists between an observation and a series of observations that have been lagging behind it. The representation of this is made up of the sum of $\theta 1(n-1) + \cdots + \theta p(n-p)$. Differences (I): the differences term is a representation of the specific amount of differencing that was done in order to render the time series stationary, d is the symbol that is used to denote it in the ARIMA (p, d, q) formula. Moving average: The moving average term is a linear function of the prior errors or residuals, and it indicates the error or residual at a specific time. The representation of this is made up of the sum of $\theta 1(n-1) + \cdots + \theta q(n-q)$. In conclusion, the ARIMA model utilizes a weighted sum of previous observations and previous forecast errors to make predictions about the future value in a time series. The weights of the ARIMA model are determined by the values of p, d, and q.

AR (Autoregression) indicates that the ARIMA model accounts for the fact that there is an autocorrelation between the present-moment observations and the observations from multiple moments prior; in other words, that the present-moment observations can be inferred from the past observations. In order to forecast future data, the AR model takes into account both the present and previous observations [9].

I (Integrated) stand for differencing, the process of smoothing down time series data so it fits the time series model's assumptions. Common methods include first- or second-order differencing, which involves comparing the initial data set twice.

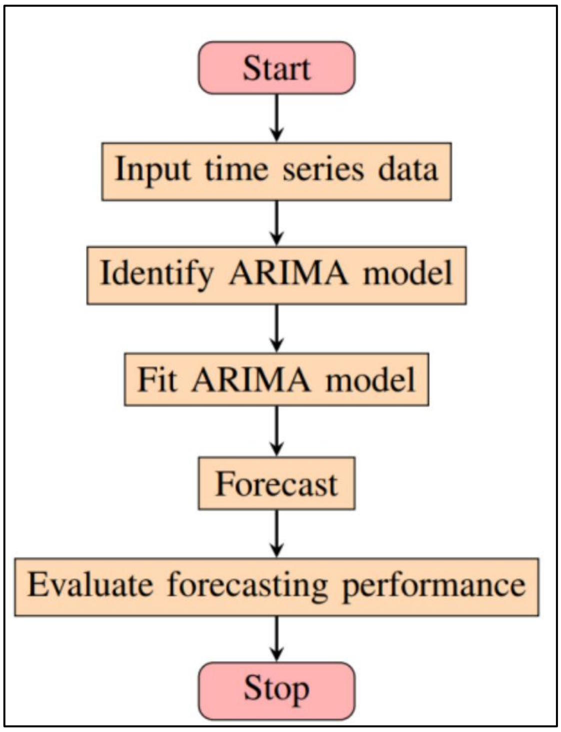


Figure 3. The flow chart of ARIMA

MA (Moving Average) which implies that in the ARIMA model, the present-moment data are correlated with the errors of the observations from many moments prior. When making predictions about future observations, the MA model takes a sliding average of its previous mistakes. [10].

Long-Short Term Memory

LSTM can process inputs from a variety of sources because it is an expansion of the original LSTM model [13]. To improve the accuracy of crime forecasts, an LSTM model can take into account a number of variables, which include including crime process. [14]. The basic premise of Long Short-Term Memory (LSTM) is that the network may selectively remember and forget things by controlling the flow of information through a gating mechanism [15]. There are three gates that make up an LSTM: the input gate, the oblivion gate, and the output gate. For managing the flow of information, each gate uses a sigmoid activation function that outputs a value between 0 and 1. For filtering and passing information, each gate employs a dot product operation. To be more precise, the input gate regulates the process by which fresh inputs are integrated into long-term memory, the forgetting gate regulates the process by which information is erased from long-term memory, and the output gate regulates the effect of long-term memory on the output at the present time. Additionally, the LSTM makes use of a cell state for data storage and transport [16]. The input and forgetting gates allow the cell state to be managed in a selective manner, allowing the user to update and forget parts of the information as needed. To make a prediction for the present moment, the LSTM updates the hidden state and cell state at each moment using the current input, the hidden state from the previous moment, and the cell state. From a mathematical perspective, we can show the multivariate LSTM by starting with the basic LSTM model and adding more variables to it. The basic LSTM model consists of three gates (input, forget, and output) and a memory cell Let x_t denote the input at time t, h_t denote the hidden state at time t, and i_t denote the cell state at time t. The imput gate i_t, forget gate f_t, and output gate o_t are defined as follows by formula (2):

$$\begin{gathered} i_t = \sigma(w_{xt}x_t + W_{hi}h_{t-1} + w_{ct}c_{t-1} + b_i) \\ f_t = \sigma\left(x_{xf}x_t + w_{hf} + w_{cf}c_{t-1} + b_f\right) \\ o_t = \sigma(w_{x0}x_t + w_{ho}h_{t-1} + w_{co}c_t + b_0) \end{gathered} \tag{2}$$

Where σ is the sigmoid function, W and b are the weight matrix and bias vector, respectively. The cell state c_t is updated as follows by formula (3):

$$c_t = f_t ct - 1 + i_t \tanh\,(w_{xc}x_t + X_{hc}h_{t-1} + b_c) \tag{3}$$

The hidden state h_t is computed as follows by formula (4):

$$h_t = o_t \tanh\ c_t \tag{4}$$

To extend this model to multiple variables, we can add an additional input vector $x_{i,k}$ for each variable $x = 1,2, \dots, K$ at time t. Then, the input gate, forget gate, and output gate become the following formula (5):

$$\begin{gathered} i_{t,l} = \sigma\left(w_{xt,k}x_{t,k} + w_{ht}h_{t-1} + w_{ci}c_{t-1} + b_i\right) \\ f_{t,k} = \sigma\left(x_{xf,k}x_{t,k} + w_{hf} + w_{cf}c_{t-1} + b_f\right) \\ o_{t,k} = \sigma\left(w_{x0,k}x_{t,k} + w_{ho}h_{t-1} + w_{c0}c_t + b_0\right) \end{gathered} \tag{5}$$

where $w_{i,k}, W_{hi}$, and $W_{x0,k}$ are the weight matrices for input, forget, and output gates, respectively, for the k-th variable.

The cell state and hidden state are updated as before by the following formula (6):

$$\begin{gathered} \sum_{k=1}^{k} f_{t,k}c_{t-1,k} + \sum_{k=1}^{K} i_{t,k} \tanh\left(W_{xc,k}x_{t,k} + W_{hc}h_{t-1} + b_c\right) \\ h_t = o_t \tanh c_t \end{gathered} \tag{6}$$

where the K^{th} variable's weight matrix is denoted by $W_{xc,k}$ and $c_{t,k}$ is the K^{th} variable's cell state at time t. Time series forecasting involving many variables can be accomplished with this multivariate LSTM model. To increase prediction accuracy and capture more complicated connections, the model can use additional variables.

Hybrid ARIMA and LSTM for Crime Prediction and Forecasting Analysis

The well production data are in the form of time series data and can be assumed to consist of a linear portion and nonlinear portion, which can be expressed as follows:

$$x_t = L_t + N_t + \varepsilon_t \tag{7}$$

where L_t denotes the linearity of the data at time t, N_t signifies nonlinearity, and ε_t denotes the error term. The ARIMA method can successfully model nonlinear relationships in the time series data, and LSTM can successfully model nonlinear components. As shown in Figure 6, we develop two hybrid models that combine the benefits of the ARIMA and LSTM approaches in order to achieve the best possible outcomes in terms of predicting. Figure 5 shows the suggested method's flowchart, which breaks down the hybrid models into four stages: (1) capturing the first data. Original production data from Indian crime data are utilized in this work. Thus, their findings can reveal changes in well production that are based on reality. [2] ARIMA model linear prediction.

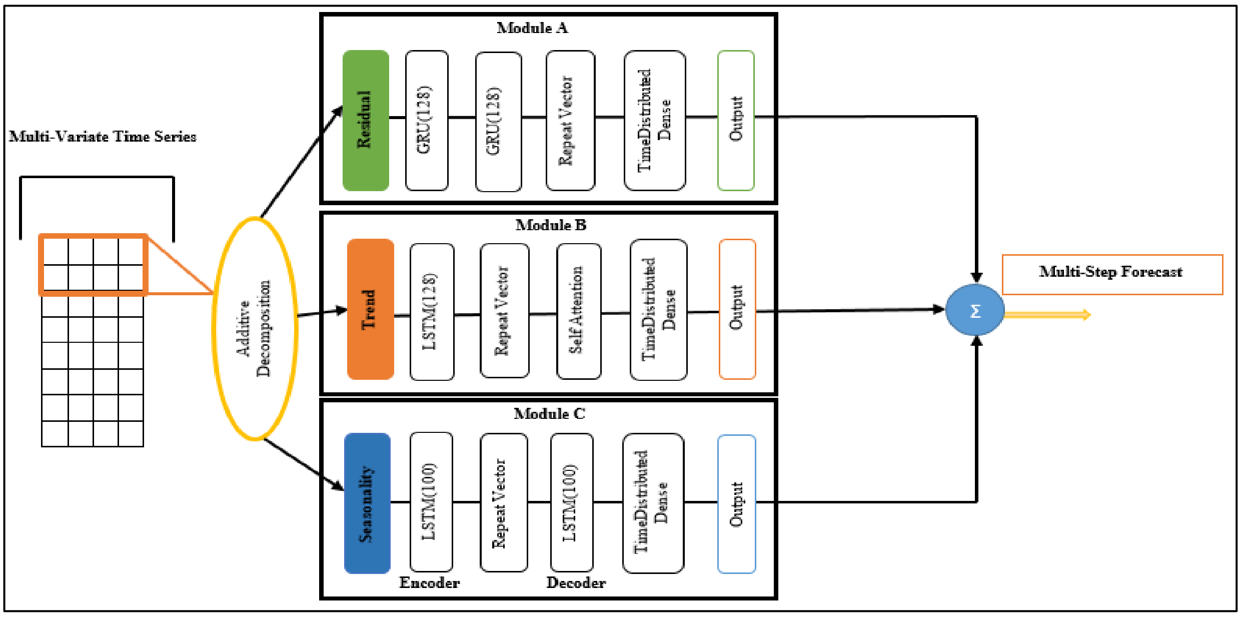


Figure 4. Proposed LSTM Architecture

To get the linear part of the production time series, which is denoted as L_t, and to get the residual items, which are input terms for the following step, the ARIMA statistical model is used. Three, LSTM modeling for nonlinear prediction. In model 1 (ARIMA-LSTM), the residuals from the ARIMA model are the only inputs for the LSTM machine learning model. Hence, we forecast the nonlinear data as $N_\varepsilon = f(\varepsilon(t-1), \varepsilon(t-2), \cdots, \varepsilon(t-p))$. In model 2 (ARIMA-LSTM-DP), the input times series data include the residual terms and daily production time series. Hence, nonlinear data forecasting can be expressed as N $N_\varepsilon = f(\varepsilon(t-1), \varepsilon(t-2), \cdots, \varepsilon(t-p); h(t-1), h(t-2), \cdots, h(t-p))$. (4) Coupling and evaluating the final result of ARIMA-LSTM model.

Algorithm 1: Training Strategy

Step 1: Applying additive decomposition, separate time series into three parts: residual, trend, and seasonality.
Step 2: For every part, run a separate train-test split.
Step 3: Build data windows for each unit to feed into the modules.
Step 4: Utilize residual train data, self-attention, and training to construct a stacked ARIMA module. An LSTM encoder decoder module with a long short-term memory (LSTM) trained on trend train data.
Step 5: Combine the outputs of each module's predictions and apply the trained models to the test data.

Rapid development has made cities crime-ridden hotspots. Figure 7 is a clustered bar graph showing the total crimes in India. In Delhi, you will find the highest crime rate. With a population of 10,980,000 and a crime score of 59.58, Delhi was ranked as the city with the greatest crime rate in a report from 2022. With no drug offenses reported on Twitter, Kolkata has the second-lowest overall crime count. This is in continuation of the 2022 NCB report that ranked Kolkata as the safest city in India. Keep in mind that the number on this graph is directly related to the number of Twitter users in each city.

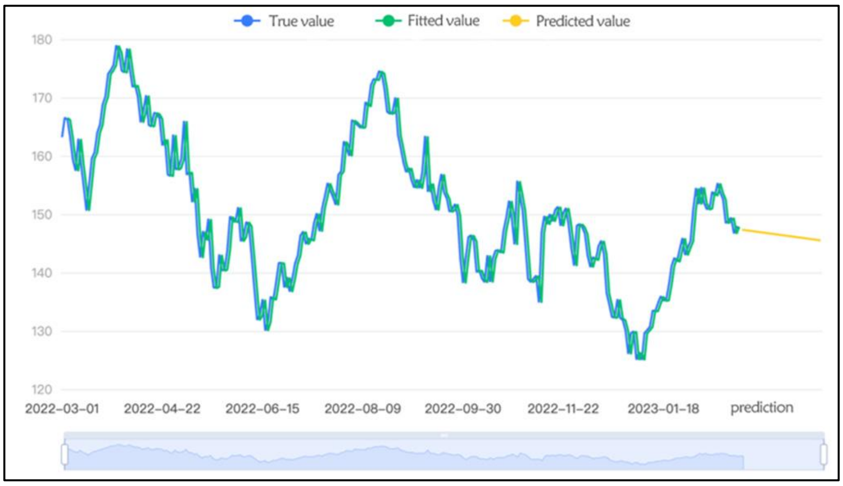


Figure 5. Proposed model for crime prediction

Algorithm 2: ARIMA-LSTM

```
Data: A set of dependent variables $y$, a forecasting horizon: $h$, and hyperparameters for ARIMA and LSTM stages
Result: Predictions of ARIMA-LSTM-CF, $y_b$
Use SDK Means++clustering algorithm to select workday data $y_a$ from raw data $y$. Analyze data characteristics to obtain feature dataset.
Take $y_a$, and $h$ as the inputs of ARIMA to obtain the prediction results of linear part $y_p^{\text{ARIMA}}$.
Calculate the residuals of ARIMA: $\varepsilon_t = y_a - y_p^{\text{ARIMA}}$
Take $\varepsilon_t$ and $h$ as the inputs of LSTM to obtain the prediction results of nonlinear part(residuals) $y_p^{LSTM}$
Calculate the predictions of ARIMA-LSTM: $y_p = y_p^{ARIMA} + y_p^{LSTM}$
```

To evaluate the prediction effect of different models, four evaluation indexes are selected to test the prediction accuracy of traffic data, including root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and R2 score.

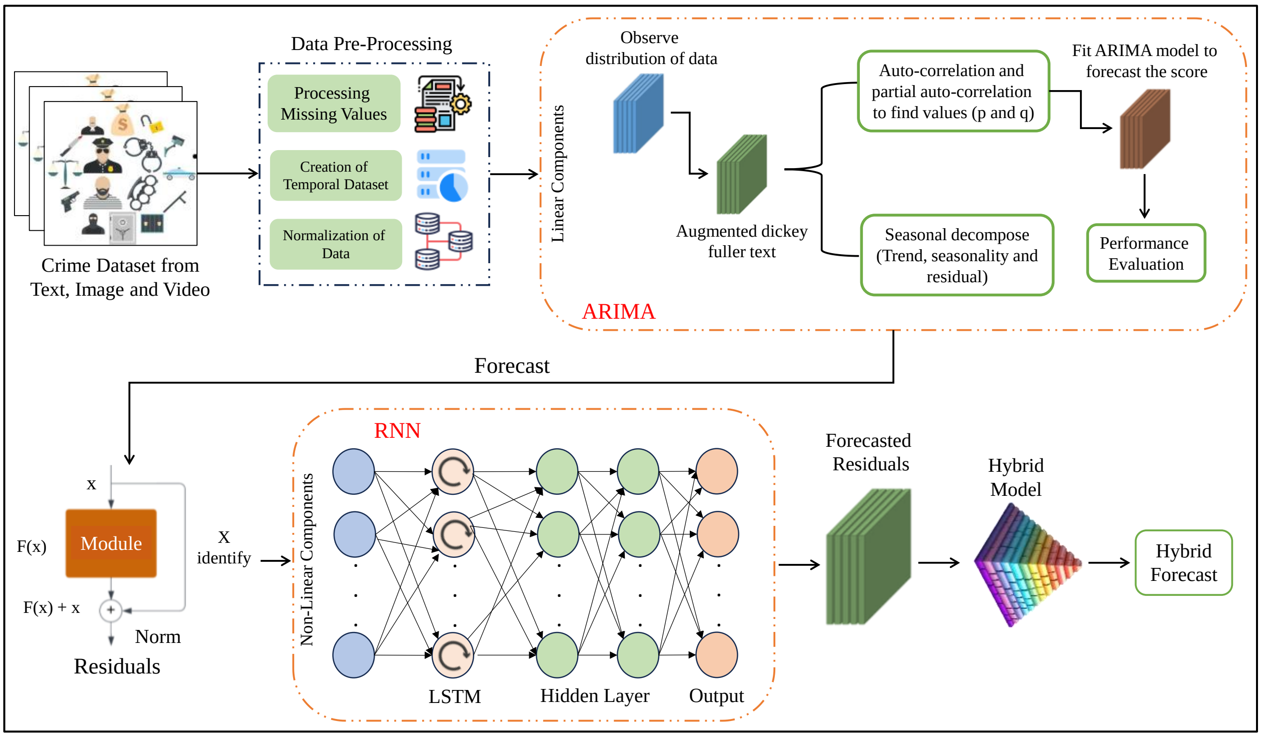

Crime Dataset from Text, Image and Video
Data Pre-Processing
Processing Missing Values
Creation of Temporal Dataset
Normalization of Data
Linear Components
Observe distribution of data
Augmented dickey fuller text
Auto-correlation and partial auto-correlation to find values (p and q)
Seasonal decompose (Trend, seasonality and residual)
Fit ARIMA model to forecast the score
Performance Evaluation
ARIMA
Forecast
x
F(x)
Module
F(x) + x
Norm
Residuals
X identify
Non-Linear Components
RNN
LSTM
Hidden Layer
Output
Forecasted Residuals
Hybrid Model
Hybrid Forecast

Figure 6. Proposed Hybrid ARIMA-LSTM for Crime Prediction and Forecasting

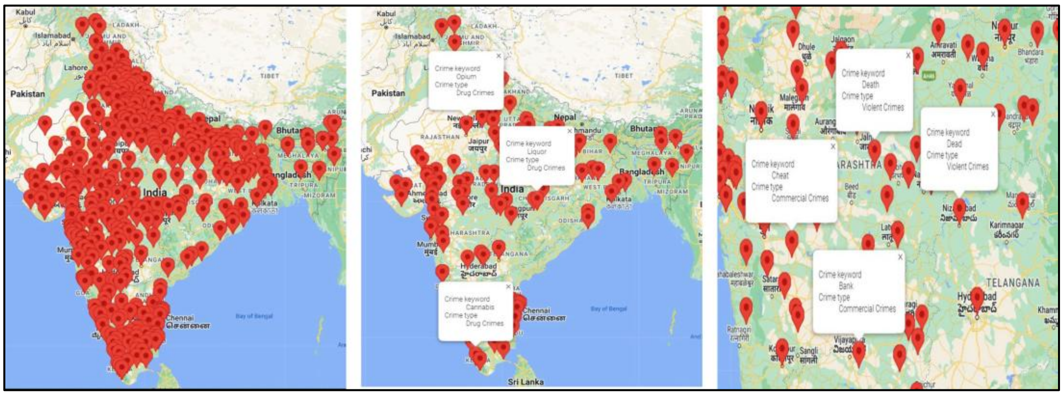

Kabul
Islamabad
Lahore
Pakistan
TIBET
Nepal
Bhutan
India
Bangladesh
Kolkata
Chennai
Bay of Bengal
Sri Lanka
Crime keyword Opium Crime type Drug Crimes
Crime keyword Liquor Crime type Drug Crimes
Crime keyword Cannabis Crime type Drug Crimes
Crime keyword Death Crime type Violent Crimes
Crime keyword Dead Crime type Violent Crimes
Crime keyword Cheat Crime type Commercial Crimes
Crime keyword Bank Crime type Commercial Crimes
TELANGANA

Figure 7. Pinpointing crimes through scatter plots

The mathematical description of these evaluation indexes is given in formulas (8) - (11).

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \left(x_i - x_i'\right)^2} \tag{8}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} \left|x_i - x_i'\right| \tag{9}$$

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n} \left|\frac{x_i - x_i}{x_i}\right| \tag{10}$$

$$R^2 = 1 - \frac{\text{SSE}}{\text{SST}} = 1 - \frac{\frac{\sum_{i=1}^{n}(x_i - x')^2}{n}}{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n}} = 1 - \frac{\text{RMSE}}{\text{Var}} \tag{11}$$

where x_i is the original value, x_i' is the prediction value, n and is the number of time-series data. SSE is the residual square sum, and SST is the total dispersion square sum.

RESULTS

The results and discussion segment are divided into four sections based on the methodology outlined in Figure 6. These sections include predictive accuracy, time series analysis using LSTM, exploratory data analysis, and forecasting using an ARIMA model. Additionally, the experimental data are displayed and analyzed in every section. We start by looking at how various algorithms compare in terms of predicted accuracy. Part two involved evaluating the model's efficacy via crime data analysis using ARIMA-LSTM. Next, we go into the specifics of the crime in the exploratory data analysis portion. Lastly, we use the ARIMA-LSTM model to demonstrate crime predictions and future trends. The results were generated using a variety of Python libraries, such as several others, including Keras with Tensor Flow, Sk Learn, Pandas, Numpy, Seaburn, and Scipy.

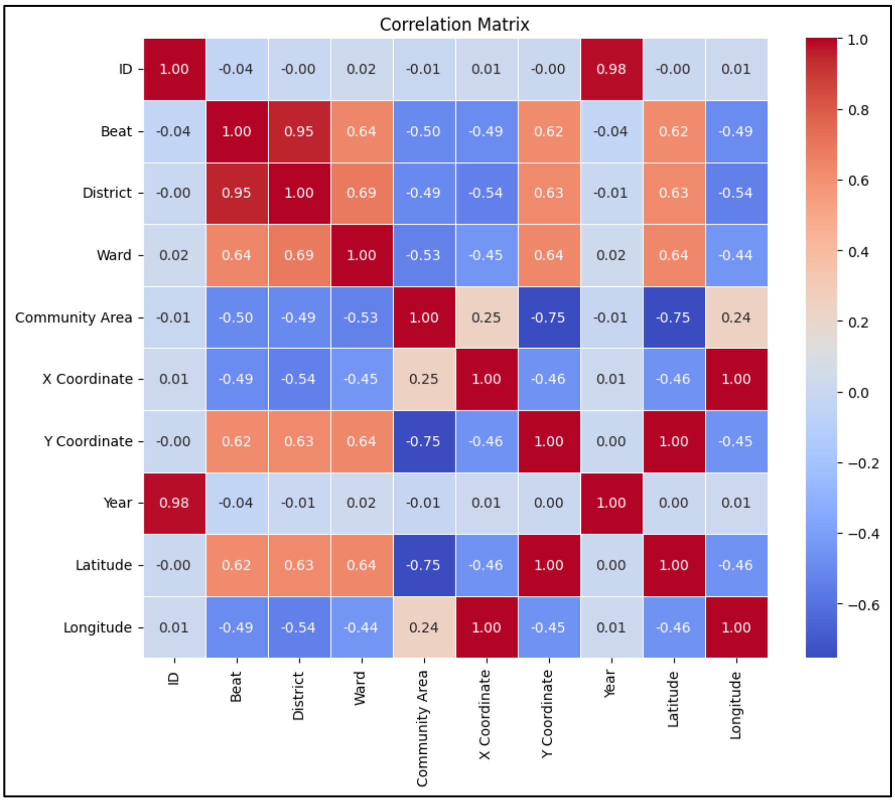


Figure 8. Confusion Matrix for proposed model

To the best of our knowledge, these techniques have not been combined for use with Indian crime statistics on any occasion. Therefore, population density—which has historically been associated with greater crime rates according to big data—is the primary factor to consider when selecting these cities. The data was pre-processed to remove noise and converted to stationary format before LSTM was applied. To facilitate handling and improve classification, time series data must be changed from their non-stationary form into a stationary one [49]. To ensure that the data is steady and to assess the correctness of the error scores, the Dickey-Fuller test is used [50]. The findings offer comprehensive recommendations derived from data processing and ARIMA-LSTM model training for a collection of time-series data. It is common practice to quantify scale-dependent error and percentage error when dealing with time-series data. In this case, the RMSE and MAE, two well-known metrics of scale-dependent error, were employed, in addition to the batch size and number of epochs. The root-mean-squared error (RMSE) quantifies the typical error size. It is the exact formula: square root of average squared discrepancies between actual observations and predictions. In cases where significant mistakes are highly undesired, the RMSE will prove to be more valuable. The MAE takes into account both the direction and magnitude of the errors in a set of forecasts. All individual differences are given equal weight in the average of the absolute differences between the expected and actual observations across the test sample. Figure 8 displays the ARIMA-LSTM confusion metrics, which show how well the associated model performed on testing data instead of training data.

The Figure 8 matrix provides insights into the model's performance, showing high precision in identifying true positives and effectively minimizing false positives and negatives. This evaluation underscores the reliability of our hybrid approach in anticipating crime trends, crucial for enhancing law enforcement strategies and policy-making efforts. The Figure 9 depicting crime rate forecasts using the hybrid ARIMA-LSTM model showcases the model's ability to handle both the temporal dependencies inherent in crime data and the long-term trends that LSTM excels at capturing. his hybridization approach not only improves forecasting precision but also offers a versatile framework applicable to various crime types and geographical contexts, thereby aiding proactive law enforcement strategies and resource allocation.

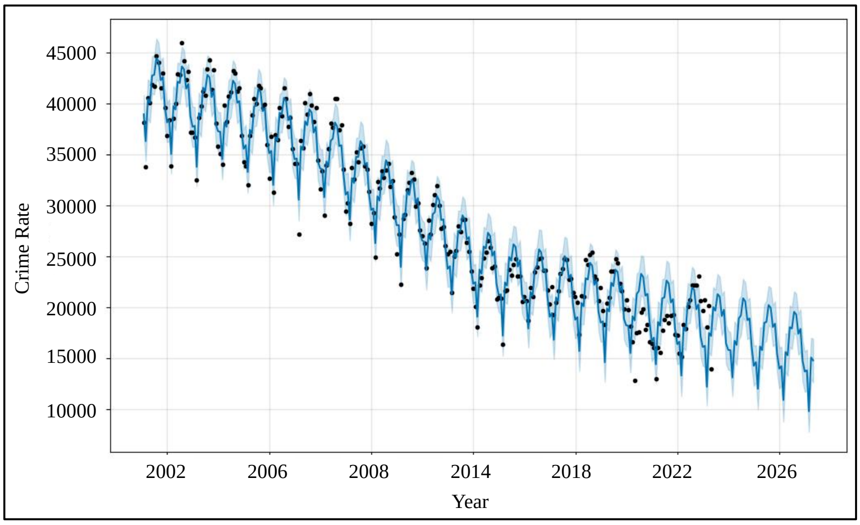


Figure 9. Forecasted Crime Rates (2002-2026) Using ARIMA-LSTM Hybrid Model

The Figure 10 illustrates the execution time across various datasets (x-axis) for different models, including LSTM, ARIMA, Bi-LSTM, and our proposed hybrid model (y-axis). This comparative analysis highlights the efficiency and computational performance of the proposed model, demonstrating its advantages in terms of reduced execution time across diverse datasets.

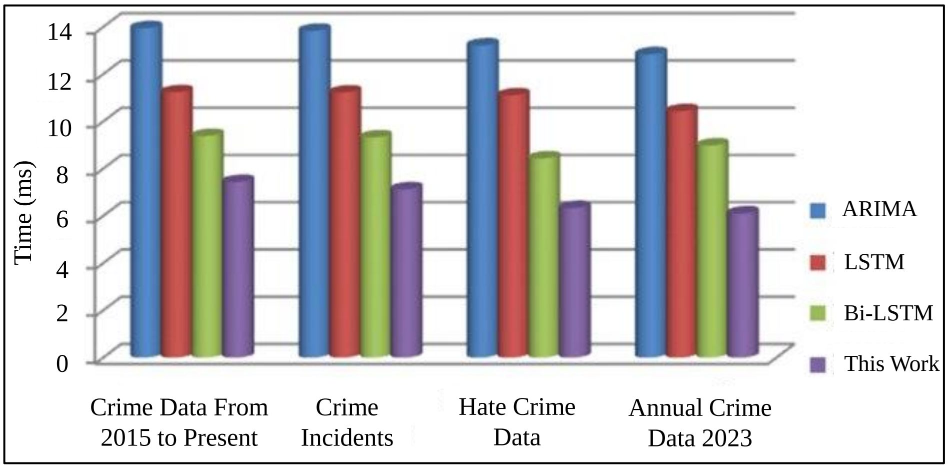


Figure 10. Representation of time across various datasets

Understanding the nuances of training and validation loss, as well as accuracy, is essential for evaluating and improving machine learning models. Training loss quantifies how well a model fits the training data by measuring the error between the model's predictions and the actual target values during training (See Figure 11). A decreasing training loss generally indicates that the model is learning effectively. However, a significant drop in training loss without a corresponding decrease in validation loss often signals overfitting, where the model captures noise and specific details from the training data that do not generalize to unseen data. Validation loss, on the other hand, provides an estimate of the model's performance on a separate validation dataset not used during training. This helps in assessing how well the model generalizes to new, unseen data.

The validation loss is calculated after each epoch, and techniques like early stopping can be employed to halt training when the validation loss starts increasing, thus preventing overfitting. Ideally, both training and validation loss should decrease and converge to low values, indicating effective learning and good generalization. A large gap between these losses suggests overfitting, while similar values suggest robust model performance. Conversely, high values for both indicate underfitting, where the model is too simplistic to capture the data's underlying patterns. Accuracy, a common evaluation metric for classification models, represents the proportion of correctly predicted instances out of the total instances. It is particularly straightforward and useful for balanced datasets. However, for imbalanced datasets, accuracy can be misleading, as it does not consider the distribution of classes. In such cases, complementary metrics like precision, recall, F1 score, and the confusion matrix should be used to obtain a comprehensive evaluation. In the context of your research, graphs illustrating the training and validation loss, as well as training and testing loss across different models, provide valuable insights into model performance. These graphs demonstrate how the proposed hybrid model outperforms others by achieving lower losses, indicating better learning and generalization.

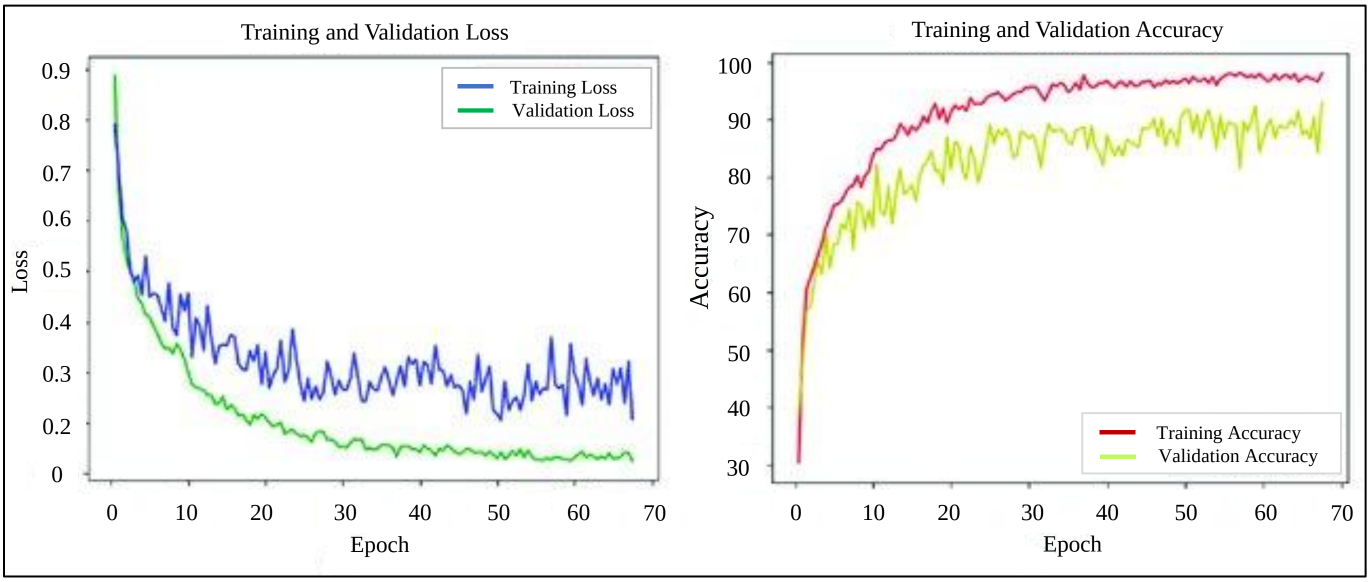


Figure 11. Training and Validation Loss and Training and Validation Accuracy Graph of proposed ARIMA-LSTM

Accurate crime prediction is crucial for public safety and law enforcement planning, and traditional statistical methods lack in prediction and forecasting (See Figure 13). The proposed ARIMA-LSTM (See Figure 14) model effectively captures linear and nonlinear trends and seasonality in crime data by combining autoregressive, integrated, and moving average components with LSTM. Representing actual versus predicted crime rates involves visualizing both on the same graph over time, allowing for the identification of trends, seasonal patterns, and deviations where predictions diverge from actual data. Performance metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2) provide quantitative measures of model accuracy and reliability. Similarly, graphs comparing execution time for LSTM, ARIMA, Bi-LSTM, and the proposed hybrid model across various datasets highlight the computational efficiency and robustness of the proposed approach. By thoroughly understanding and analysing training and validation loss alongside accuracy, and utilizing a combination of evaluation metrics, you can ensure a balanced and robust assessment of your model's performance, ultimately leading to more reliable and effective predictive systems.

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

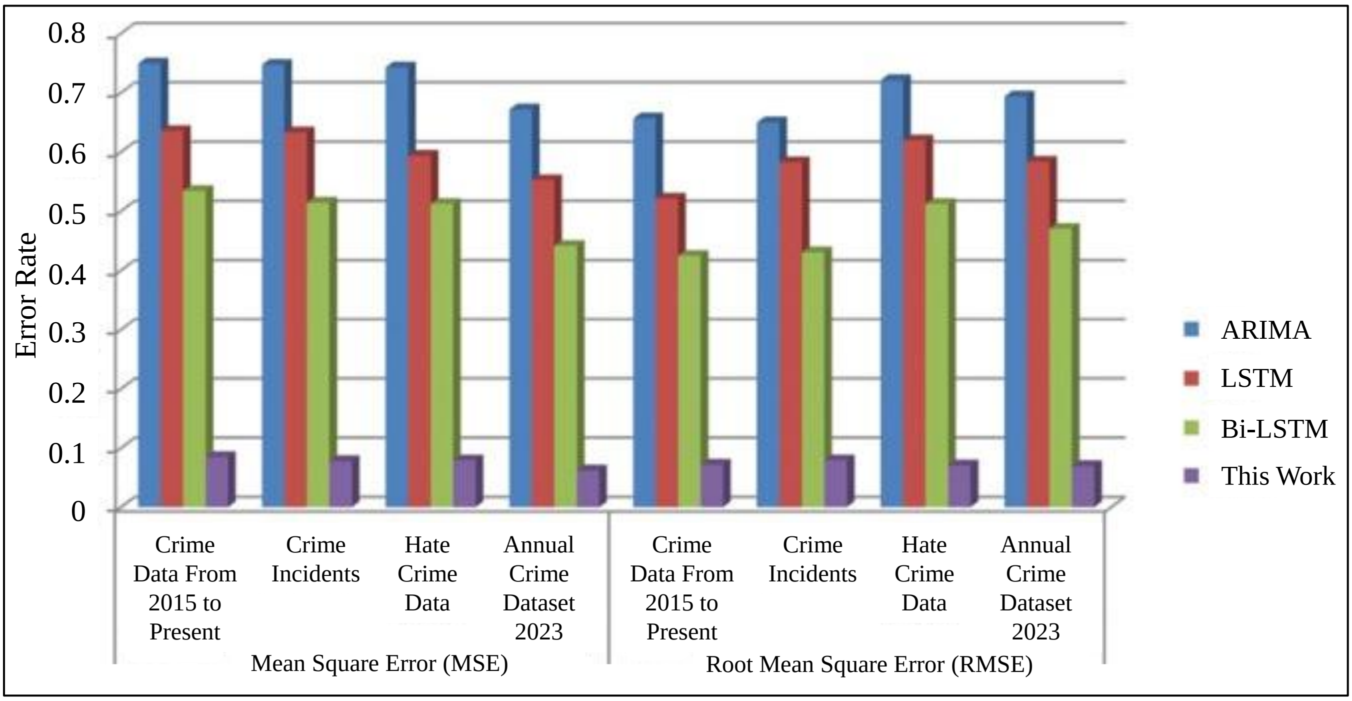


Figure 12. Representation of error rate for MSE and RMSE on proposed approach

The integration of AutoRegressive Integrated Moving Average (ARIMA) models with Recurrent Neural Networks (RNNs) represents a significant advancement in the domain of time series forecasting, particularly in the context of crime prediction. This hybridization strategy is rooted in the recognition that while ARIMA models excel at modeling linear temporal dependencies and effectively capturing underlying trends and seasonality in sequential data, they often fall short in handling complex nonlinear patterns that are intrinsic to real-world datasets like crime records. Conversely, RNNs, especially advanced variants such as Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs), have demonstrated exceptional capabilities in learning long-range dependencies and modeling nonlinear, temporal relationships within time series data. However, when used in isolation, RNNs may require extensive training data and computational resources to fully capture both the linear and nonlinear aspects of a dataset.

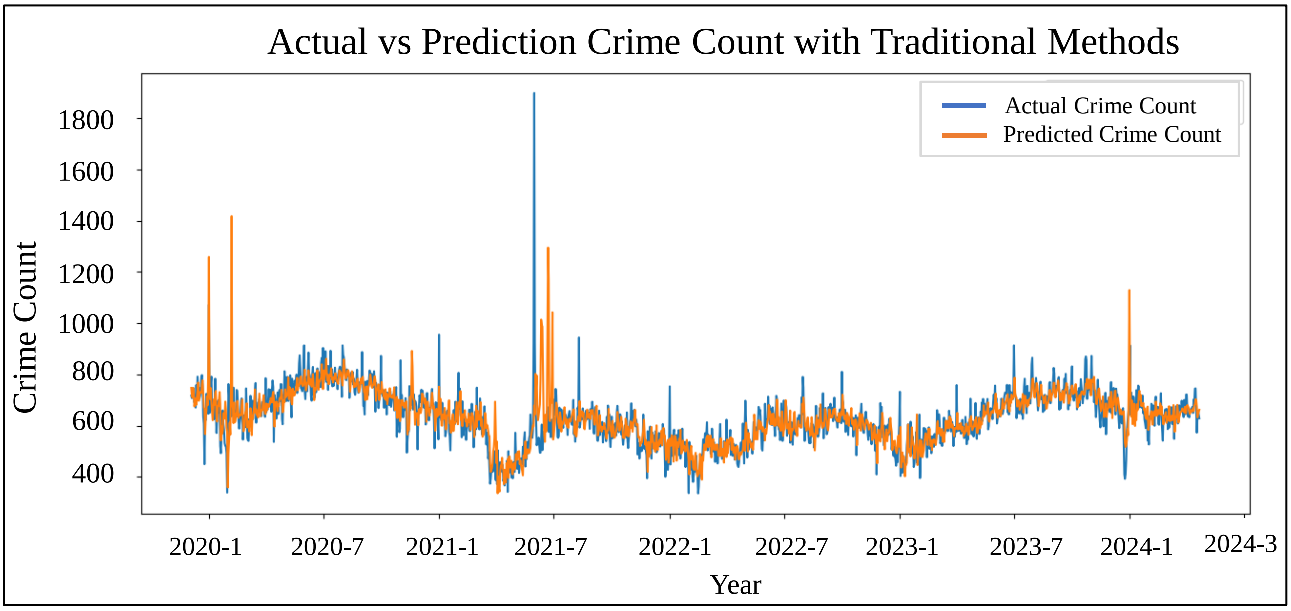


Figure 13. Represents actual vs predicted crime rate using traditional methods

Journal of Information Systems Engineering and Management
2025, 10(38s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

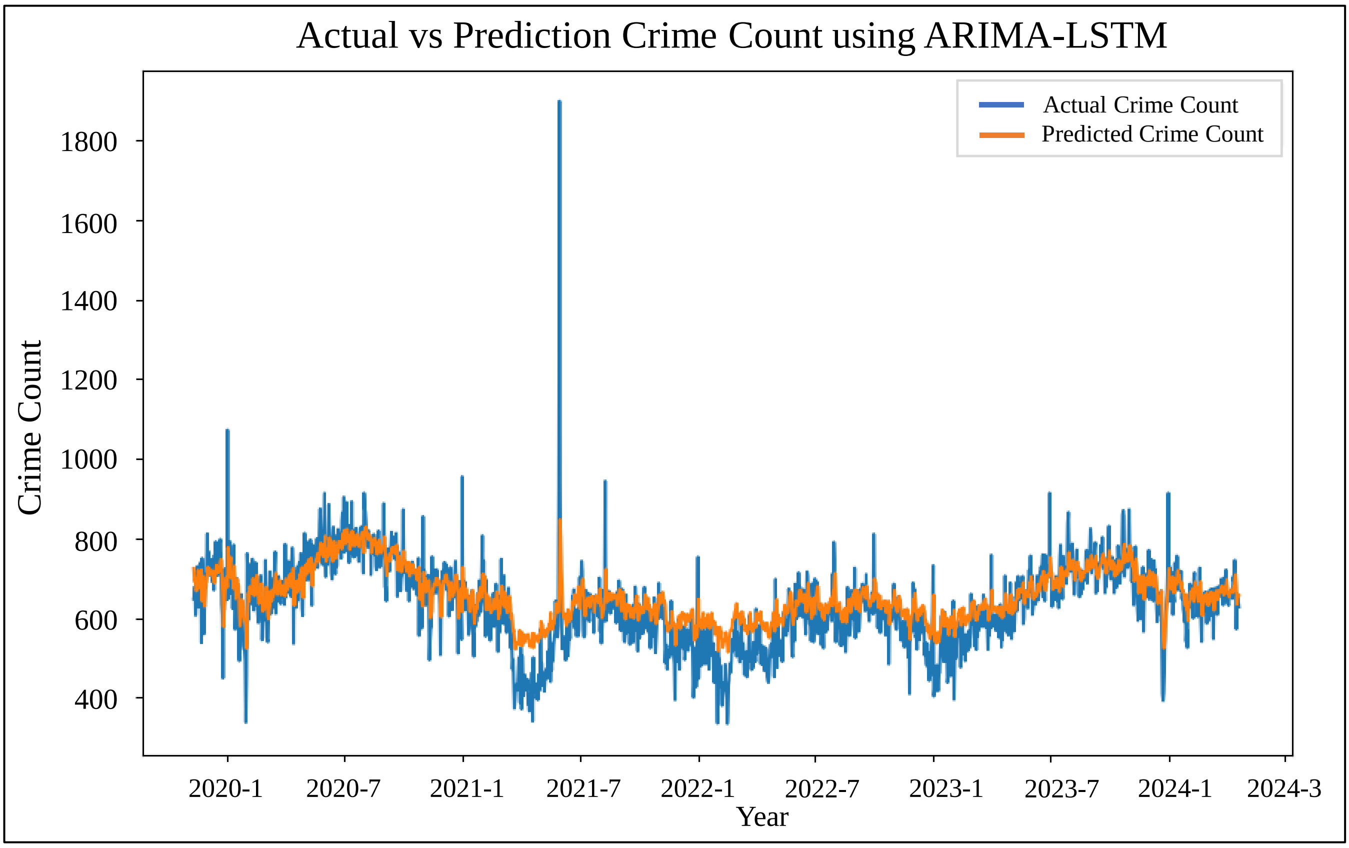


Figure 14. Represents actual vs predicted crime rate using proposed work

By combining the strengths of these two complementary modeling approaches, the proposed hybrid ARIMA-RNN model provides a robust framework for crime forecasting. Initially, ARIMA is employed to model and remove the linear components such as trend and seasonality from the crime data, thereby isolating the residuals that predominantly contain the nonlinear and stochastic elements of the time series. These residuals, which ARIMA alone cannot effectively capture, are then used as input for the RNN model. The RNN subsequently learns the complex nonlinear patterns embedded within these residuals, effectively modeling the dynamics that ARIMA fails to address. This staged approach ensures that each component of the hybrid model focuses on the type of pattern it is best suited to learn, thereby enhancing the overall predictive capability.

Through extensive experimentation using real-world crime datasets collected across various geographic regions and crime categories in India, this study validates the efficacy of the hybrid model. Comparative analyses against standalone ARIMA and RNN models reveal that the hybrid approach consistently delivers superior performance across multiple evaluation metrics, including Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE). These results highlight the model's ability to generalize well across different contexts and its robustness in capturing both seasonal and irregular patterns in crime data.

Moreover, the findings from this research contribute meaningfully to the growing body of literature on hybrid forecasting models by empirically demonstrating that the fusion of classical statistical techniques and modern deep learning approaches can outperform either methodology when applied in isolation. This novel ARIMA-RNN fusion framework underscores the potential of hybrid methodologies in addressing complex predictive tasks, especially in fields like criminology, where data often exhibits both structured periodic trends and chaotic, nonlinear variations. The success of this hybrid model paves the way for future research exploring similar integrative frameworks in other application domains, thereby advancing the broader field of time series analysis and predictive modeling.

CONCLUSION

In this study, crime rate prediction was approached using two advanced neural network models: ARIMA and LSTM. Despite their inherent complexity, the outcomes were notably promising. The findings indicated that both ARIMA and LSTM models outperformed a traditional baseline model in capturing the trends and variations of crime rates. This superiority can be attributed to their robust capabilities in generating more precise predictions compared to conventional methods. The rationale behind adopting the ARIMA and LSTM models lies in their capacity to provide enhanced accuracy. Furthermore, the incorporation of multiple LSTM layers contributed to reducing the training time, making the models more efficient in handling large datasets and complex temporal dependencies inherent in crime data. Overall, the proposed approach utilizing ARIMA and LSTM represents a sophisticated and effective methodology for modelling, analysing, and predicting crime rates. By leveraging the strengths of deep learning alongside traditional time series techniques, the study underscores the potential of hybrid models in advancing the field of crime prediction and forecasting.

REFRENCES

[1] Dakalbab F, Abu Talib M, Elmutasim O, Bou Nassif A, Abbas S, Nasir Q (2022) Artificial intelligence & crime prediction: a systematic literature review. Soc Sci Humanit 6:100342.

[2] Butt, U. M., Letchmunan, S., Hassan, F. H., & Koh, T. W. (2022). Hybrid of deep learning and exponential smoothing for enhancing crime forecasting accuracy. Plos one, 17(9), e0274172.

[3] Safat, W., Asghar, S., & Gillani, S. A. (2021). Empirical analysis for crime prediction and forecasting using machine learning and deep learning techniques. IEEE access, 9, 70080-70094.

[4] S. Chackravarthy, S. Schmitt and L. Yang, "Intelligent crime anomaly detection in smart cities using deep learning", Proc. IEEE 4th Int. Conf. Collaboration Internet Comput. (CIC), pp. 399-404, Oct. 2018.

[5] Jamii, J., Trabelsi, M., Mansouri, M., Kouadri, A., Mimouni, M. F., & Nounou, M. (2024). Medium-term wind power forecasting using reduced principal component analysis based random forest model. Wind Engineering, 0309524X231217912.

[6] A. Fidow, M. Hassan, M. Imran, X. Cheng, C. Petridis and C. Sule, "Suggesting a hybrid approach mobile apps with big data analysis to report and prevent crimes", Social Media Strategy in Policing, pp. 177-195, 2019.

[7] X. Zhang, L. Liu, L. Xiao and J. Ji, "Comparison of machine learning algorithms for predicting crime hotspots", IEEE Access, vol. 8, pp. 181302-181310, 2020.

[8] G. R. Nitta, B. Y. Rao, T. Sravani, N. Ramakrishiah and M. BalaAnand, "LASSO-based feature selection and Naïve Bayes classifier for crime prediction and its type", Service Oriented Comput. Appl., vol. 13, no. 3, pp. 187-197, Sep. 2019.

[9] A. L'Heureux, K. Grolinger, H. F. Elyamany and M. A. M. Capretz, "Machine learning with big data: Challenges and approaches", IEEE Access, vol. 5, pp. 7776-7797, 2017.

[10] Z. Zhang, D. Sha, B. Dong, S. Ruan, A. Qiu, Y. Li, et al., "Spatiotemporal patterns and driving factors on crime changing during Black Lives Matter protests", ISPRS Int. J. Geo-Inf., vol. 9, no. 11, pp. 640, Oct. 2020.

[11] W. Safat, S. Asghar and S. A. Gillani, "Empirical Analysis for Crime Prediction and Forecasting Using Machine Learning and Deep Learning Techniques," in IEEE Access, vol. 9, pp. 70080-70094, 202.

[12] S. H. Amin Mahmood and A. Abbasi, "Using Deep Generative Models to Boost Forecasting: A Phishing Prediction Case Study," 2020 International Conference on Data Mining Workshops (ICDMW), Sorrento, Italy, 2020, pp. 496-505.

[13] Tekin, S.F., Kozat, S.S. Crime prediction with graph neural networks and multivariate normal distributions. SIViP 17, 1053–1059 (2023).

[14] Salinas, D., et al.: DeepAR: probabilistic forecasting with autoregressive recurrent networks. Int. J. Forecast. 36(3), 1181–1191 (2020).

[15] Wang, B., Luo, X., Zhang, F., et al.: Graph-Based Deep Modeling and Real Time Forecasting of Sparse Spatio-Temporal Data (2018).

[16] Stec, Alexander, and Diego Klabjan. "Forecasting Crime with Deep Learning." ArXiv, (2018).

[17] A. Ghazvini, S. N. H. S. Abdullah, M. Kamrul Hasan and D. Z. A. Bin Kasim, "Crime Spatiotemporal Prediction with

[18]Fused Objective Function in Time Delay Neural Network," in IEEE Access, vol. 8, pp. 115167-115183, 2020.

[19]V. Pinheiro, V. Furtado, T. Pequeno, and D. Nogueira, "Natural language processing based on semantic inferentialism for extracting crime information from text," in Proc. IEEE Int. Conf. Intell. Secur. Informat., Vancouver, BC, Canada, May 2010, pp. 19–24.

[20] B. Wang, P. Yin, A. L. Bertozzi, P. J. Brantingham, S. J. Osher, and J. Xin, "Deep learning for real-time crime forecasting and its ternarization," Chin. Ann. Math., B, vol. 40, no. 6, pp. 949–966, Nov. 2019.

[21] C. Rajapakshe, S. Balasooriya, H. Dayarathna, N. Ranaweera, N. Walgampaya and N. Pemadasa, "Using CNNs LSTMs and Machine Learning Algorithms for Real-time Crime Prediction," 2019 International Conference on Advancements in Computing (ICAC), Malabe, Sri Lanka, 2019, pp. 310-316.

[22] Sreeram, L., & Sayed, S. A. (2024). Short-term Forecasting Ability of Hybrid Models for BRIC Currencies. Global Business Review, 25(3), 585-605.

[23] Bukhari, A. H., Raja, M. A. Z., Alquhayz, H., Almazah, M. M., Abdalla, M. Z., Hassan, M., & Shoaib, M. (2024). Predictive analysis of stochastic stock pattern utilizing fractional order dynamics and heteroscedastic with a radial neural network framework. Engineering Applications of Artificial Intelligence, 135, 108687.

[24] Wang, H. C., & Chiu, W. P. (2017). A Hybrid Arima Elman Artificial Neural Networks Approach With Overlapped Moving Window-An Experiment Study For Crime Rate Predicting. Communications of the ICISA, 18(1), 14-41.

[25] Stephen, S., Argwings, O., & Julius, K. (2022). Application of ARIMA, hybrid ARIMA and Artificial Neural Network Models in predicting and forecasting tuberculosis incidences among children in Homa Bay and Turkana Counties, Kenya. medRxiv, 2022-07.

[26] Yin, J. (2023). Crime Prediction Methods Based on Machine Learning: A Survey. Computers, Materials & Continua, 74(2).

[27] De Oliveira, J. F., & Ludermir, T. B. (2016). A hybrid evolutionary decomposition system for time series forecasting. Neurocomputing, 180, 27-34.

[28] Mithoo, P., & Kumar, M. (2023). Social network analysis for crime rate detection using Spizella swarm optimization based BiLSTM classifier. Knowledge-Based Systems, 269, 110450.

[29] Siamba, S., Otieno, A., & Koech, J. (2023). Application of ARIMA, and hybrid ARIMA Models in predicting and forecasting tuberculosis incidences among children in Homa Bay and Turkana Counties, Kenya. PLOS digital health, 2(2), e0000084.

[30] Sreeram, L., & Sayed, S. A. (2024). Short-term Forecasting Ability of Hybrid Models for BRIC Currencies. Global Business Review, 25(3), 585-605.

[31] Popirlan, C. I., Tudor, I. V., & Popirlan, C. (2023). Predicting the unemployment rate and energy poverty levels in selected European Union countries using an ARIMA-ARNN model. PeerJ Computer Science, 9, e1464.

[32] Popirlan, C. I., Tudor, I. V., & Popirlan, C. (2023). Predicting the unemployment rate and energy poverty levels in selected European Union countries using an ARIMA-ARNN model. PeerJ Computer Science, 9, e1464.