RESEARCH ARTICLE OPEN ACCESS

Social Media Information Enriched Multimodal Diversified Deep Crime Detection Network for Crime Prediction

Sundari .P[1], Dr. Malathi. A [2]
Assistant Professor, PG and Research department of Information Technology, Govt. Arts College(A) Coimbatore- 641018, India.
Associate Professor, PG and Research Department of Computer Science, Govt. Arts College(A), Coimbatore - 641018, India.

ABSTRACT
Crime detection is crucial but challenging due to evolving criminal patterns. Social media's rise in sharing information is limited by lack of semantic understanding in performance. So, Social Media Information Enriched Multimodal Deep Crime Detection Network (SMDCnet) was developed integrated text vectors from Latent Dirichlet Allocation (LDA) and images and videos features from Mask R-CNN. These features were fed into Convolutional Bidirectional Long Short Term Memory (ConvBiLSTM) model for crime prediction. But, Bi-LSTM suffers from information redundancy and gradient explosion issues when processing diverse multimodal data. In this paper, ConvBiLSTM in SMDCnet can be improvised in two different steps. Initially, an enhanced recursive self-attention mechanism (ERSAM) is introduced in BiLSTM which integrates recursive function and Multi-Scale Self-Attention (MSSA) to solve the information redundancy issues in multi-modal data. In BiLSTM, recursion is performed by bidirectional processing which effectively compresses feature representation by analyzing the data in both forward and backward directions. BiLSTM with recursive operation captures hierarchical features through stacked layers, enhancing representation across time steps. Although the bidirectional process enhances representation without adding parameters, the doubled forward pass introduces training and inference overhead to capture context from both past and future. MSSA in BiLSTM employs attention heads with variable scales to compute feature similarity maps, improving multi-level feature relationships. Larger scales provide broader context, while smaller ones focus on local details. To optimize performance, an approximation method using MSSA across recursive layers is suggested which minimizes the cost without sacrificing accuracy. Then, loss function of ConvBiLSTM is enhanced by using convex function information entropy (CFIE). CFIE improves optimization and accelerates fitting with limited crime data. This approach prevents gradient explosion issues caused by information redundancy in Bi-LSTM and improves accuracy for more effective crime prediction. The complete proposed model is termed as Social media information enriched Multimodal Diversified Deep Crime detection network (SMDDCnet). Finally, extensive experiments demonstrate that the SMDCnet model achieves 97.54% accuracy on the Crime in India dataset, surpassing other crime prediction models.

Keywords: Crime Detection, Sentiment Analysis, Latent Dirichlet allocation, Twitter-Specific Tokenizer, Feature Pyramid Network

---

Date of Submission: 25-02-2025 Date of acceptance: 05-03-2025

I. INTRODUCTION

Crime is a significant issue in society, with daily incidents leaving citizens uneasy. Crime patterns are constantly evolving, making it difficult to analyze and predict behaviors [1]. Law enforcement agencies collect crime data using information technology (IT), but predicting crime remains challenging due to factors such as poverty and employment, which impact crime rates [2]. Crime occurrences are neither consistent nor random and the increasing number of crimes further complicates prediction efforts [3].

Traditional crime prediction methods use historical socio-economic data and demographic information from hot-spot maps to forecast different crime types [4]. However, these maps don't always accurately reflect all crimes, such as in cases like taxicab robberies where victims are dispersed across multiple locations. Additionally, the lack of generalizable data across regions hinders prediction models, as these methods primarily rely on past crime records, ignoring the socio-behavioral data from communities [5].

Public conversations and postings on social web communities like Instagram, Facebook, and Twitter provides a new form of informative data that people express their feelings and opinions. [6]. This user-generated content provides socio-behavioral signals that can be analyzed for crime prediction [7]. Sentiment Analysis (SA) or opinion mining, extracts and classifies subjective information from unstructured text, identifying the emotional tone using contextual word clues [8]. It has become a valuable tool for law enforcement to gauge public sentiment on crime by analyzing social media posts, news and reviews [9]. SA aids crime prediction by categorizing text as positive, negative, or neutral, offering insights into public safety perceptions for better resource allocation [10]. However, SA struggles with high-dimensional crime data, lexical diversity and dataset imbalances, which can limit its accuracy.

To address these challenges, Deep Learning (DL) models, a form of Artificial Intelligence (AI), have been introduced. DL algorithms can enhance crime detection and prediction by analyzing surveillance footage and classifying criminal activities such as theft, mischief and assault [11]. It can also be also integrated with smart prediction technologies like drones and Internet of Things (IoT) sensors for improved monitoring [12]. Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks are some of the DL models that are applied to analyze the spatial and temporal crime data to accurately forecast crime patterns and hotspots in cities [13].

Several DL models have been developed to reduce crime and predict trends for earlier intervention. Amongst, Multimodal DL crime prediction models leverage tweets and historical crime data to build predictive models. ConvBiLSTM [14] model extracts independent vectors from tweet and crime data, combining them to capture meaningful information. Word embedding vectors summarize this data through convolution and pooling layers. However, these models still face challenges, such as a lack of semantic understanding, which can impact prediction accuracy.

For this reason, SMDCnet [15] was developed to resolve above-mentioned issues for efficient crime prediction model. This model combines multimodal social media data with crime datasets. Text vectors are built using Latent Dirichlet Allocation (LDA) to detect crime-related topics like theft and homicide. Features from Twitter images and videos are extracted using Mask Regional CNN (MASK-RCNN) with Multi-Feature Pyramid Network (MFPN) and Non-Maximum Suppression (NMS) to enhance prediction. ConvBiLSTM [14] improves crime prediction accuracy by integrating text and image features, capturing long-term context and leveraging sentence-level semantics. But, in ConvBiLSTM, CNN extracts local features efficiently but struggles with sequential correlations. while Bi-LSTM captures bidirectional context but can't parallelize feature extraction. Hence, these issues are resolved by combining CNN and Bi-LSTM. Still, Bi-LSTM can suffer from information redundancy and gradient explosion when processing diverse multimodal data (text, images and structured data).

Hence, in this paper, Social Media Information Enriched Multimodal Diversified Deep Crime Detection Network (SMDDCnet) was developed to address the issue of BiLSTM in crime prediction. Two key enhancements are proposed in this method to improve BiLSTM performance like ERSAM and an improved loss function using CFIE to reduce information redundancy and gradient explosion. In ERSAM, recursion operation in BiLSTM is performed through bidirectional process which effectively compresses feature representation by analyzing the data in both forward and backward directions. BiLSTM with recursive function captures hierarchical feature structures through stacked layers, improving representation across time steps. This bidirectional task enhances representation without adding parameters, the doubled forward pass introduces training and inference overhead to capture context from both past and future. MSSA is applied in BiLSTM in which each attention heads with variable scales to focus on different feature regions. This assists to compute feature similarity maps efficiently and captures both global and local feature relationships. Large-scale features provide broader context and smoother results, while small-scale features highlight local details and sharp features. MSSA is applied across recursive layers using an approximation approach, which reduces computational cost without sacrificing accuracy. Then, an enhanced Loss Function using CFIE enhances the optimization and convergence speed, especially with limited crime data. This approach prevents gradient explosion by stabilizing the training process and makes the model more reliable for crime prediction.

The rest of the paper is arranged as follows, In Section II, many works linked to the crime classification and identification models are presented. The proposed SMDDCnet model is described in Section III and its validity is shown in Section IV. In Section V, the model's summary and forthcoming improvement are provided.

II. LITERATURE SURVEY

Zhou et al. [16] suggested an Unsupervised Domain Adaptation Classifier, or UDAC to detect the likelihood of crime in different cities. For every target city grid, this methodology finds a matching source city grid and uses auxiliary contexts to align the city's contexts. For precise criminal risk prediction, a Dense Convolutional Network (DCN) with UDAC learns features that are domain-invariant and uses high-level representations. But, this model encounters situational discrepancies across the source and target cities, as well as significant data inadequacy issues.

Using Twitter data and ML models, Vivek and Prathap [17] built a spatio-temporal crime detection system. The purpose of collecting this dataset was to use the tweepy module's search function to extract relevant tweets from Twitter. After then, in order to improve the data integrity, the incorrect tweets were removed using data cleaning. Last but not least, the LSTM model was used to ascertain the criminal tweet count time series forecasting for crime detection. However, poorer accuracy rates were caused by the predictive decision-making process that did not take tweet content into account.

Rayhan and Hashem [18] presented an Attention based Interpretation Spatio-Temporal model (AIST) for crime prediction. The adaptive spatio-temporal relationship correlations were applied to analyze the crime classes using the external factors such as crime vehicular movement and location data, repeated crime patterns and real crime records. The characteristics were inputted into AIST in order to capture the complex and dynamic and non-sequentialconnections of environmental reliance and temporal aspects for predicting a certain type of crime. Overfitting problems have merged as a result of insufficient data interpretation.

An MCN using crisscross optimization (CCO) on the SA was constructed by Singuluri et al. [19] for the purpose of cyber-crime prediction. By combining the outputs of both networks to optimize their strengths and boost the detection rate, the approach combines MCN with Multilayer Perceptron (MLP) using a rule based strategy. Next, the hyper-parameters of the capsule network were optimized and enhanced for crime detection using the CCO approach. Unfortunately, F1-Score and training time were both negatively affected by this model.

Escobar et al. [20] created an Agent-Based Model (ABM) for law enforcement using crime trend prediction. The technique predicts crime trends by examining offender behavior, escape trajectories, and stealing frequency. It detects crime trends and pinpoints conflicting areas for safety enhancement. The approach also creates defender positions and crime factors based on environmental data, enabling for more efficient patrol sites to combat city crime. However, significant computer resources and complex emotional models were needed to enhance the accuracy values.

Hashi et al. [21] developed a transfer learning based CNN for crime perceptive prediction by entity detection. The gathered information was pre-processed and fed into VGG-19, ResNet and GoogleNet to forecast the crime scenarios. Also, the YOLO was merged to detect the entities in support to crime prediction. However, optimizing algorithms were necessitated to refine the pre-trained CNN's parameter causing uncertainty issues and lower accuracy performance.

Mithoo & Kumar [22] presented a Spizella Swarm Optimization based Bidirectional LSTM (SSO-BiLSTM) using twitter data for crime rate detection. The pre-processed and augmented data were inputted to BiLSTM to forecast the crime patterns in related to time period. The hyper-parameter of BiLSTM were fine-tuned by SSO for convergence enhancements and lowering the models complexity. But, the models performance was hindered due to limited training data which restricts the accuracy and sensitivity rate.

Butt et al. [23] constructed a Transfer Learning (TL) with BiLSTM for crime prediction. In this model, this model reviews statistical modeling techniques for time series prediction by Simple Moving Averages (SMA), Weighted Moving Averages (WMA) and Exponential Moving Averages (EMA). Then, DL models like LSTM, BiLSTM and CNN-LSTM for analyzing time-series data. Finally, BiLSTM with TL addresses data and training challenges, improving efficiency and reducing resource and time needs for crime prediction. But, this model fails to account for temporal effects like trends, lags and periodicity.

Selvan & Sivakumaran, [24] utilized Bi-LSTM model to anticipate the criminal activities and forecast high-risk crime regions in the city. This model uses machine learning (ML) to forecast crime spots and DL to verify the alignment among predicted and actual crime incidents for analyzing crime incident data. ML algorithms process voice-based emotion data for detection, while DL methods like convolutional stacked bidirectional LSTM to handle crime scene data like audio/video, geographic coordinates and timestamps. But, this model struggled with capturing contextual information and word relationships.

Zhou et al. [25] created a Hybrid Dynamic Multi-Perspective Graph Neural Network (HDM-GNN) to detect the crime actions. This approach leverages Spatio and temporal interactions using varied urban data and incorporates the inter-regional relations across various perspectives. The compressive spatial trends and extensive temporal interactions were obtained using the Gated CNN and Graph Attention model. But, this model struggles with training spatiotemporal features from diverse sequences and effectively fusing complementary features.

III. PROPOSED SYSTEM ARCHITECTURE

This part illustrates the whole structure of the proposed SMDDCnet model and Figure 1 represents the proposed model's pipeline.

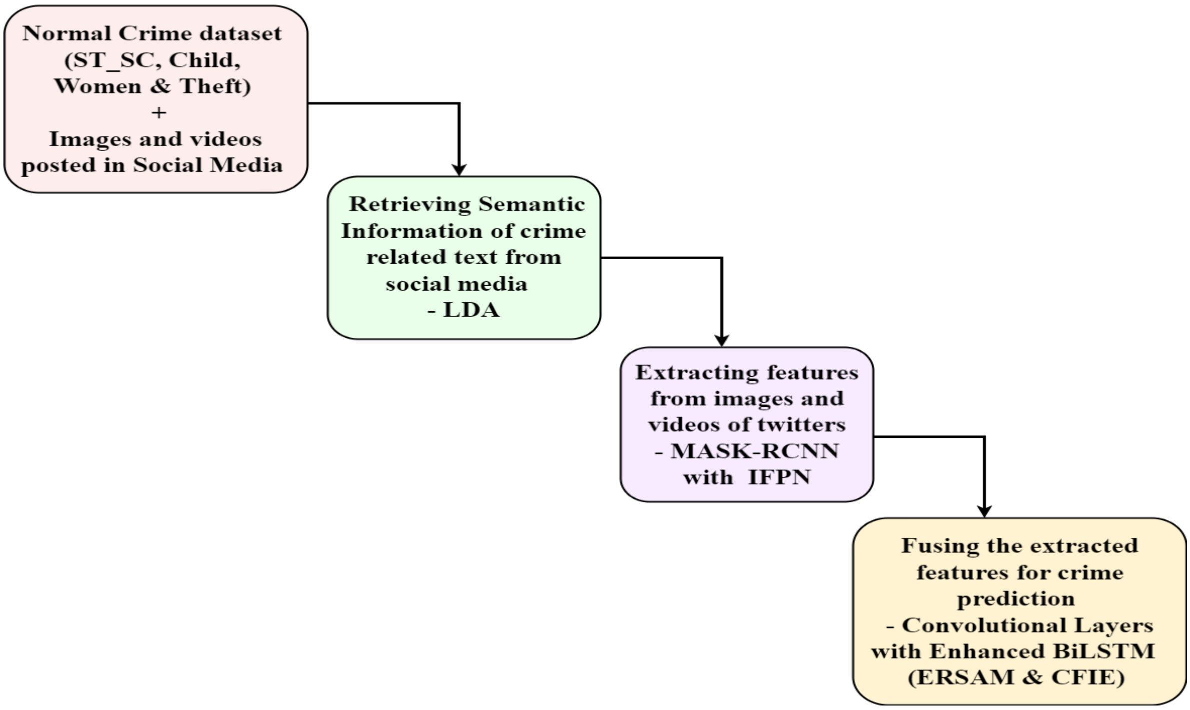

Figure 1 Pipeline of the Proposed Model

3.1 Dataset Specifications

The dataset used in this framework is "Crime in India" [26], which has all the necessary details on many facets of crimes carried out in India during 2001. This dataset allows one to investigate various elements. This dataset enables the people to have better knowledge about Indian crime statistics. There are forty-three parts of crimes from India in this collection. Few statistics provide district-level data, including police departments and special police agencies, which could vary from revenue districts. Most of the data falls between 2001 and 2010; other files provide information from 2011 and 2001–14. For the experimental purposes, four important crime classes are considered i.e., "ST_SCcrime", "Childcrime", "Womencrime" and "Theftcrime". In addition to this, image and video data associated with crime terminologies from the social media tweets of the above listed four crime classes are taken. By matching each crime data with image and video details, totally 9794 crime instances are determined for the experiment.

3.2 Model Overview

The SMDDCnet model is organized into three main modules. First, it retrieves semantic information related to crime from text using LDA. Next, it extracts features from Twitter images and videos using Mask R-CNN. This model utilizes Enhanced Recursive Self-Attention Mechanism (ERSAM) and Convex Function Information Entropy (CFIE) to enhance the efficiency of ConvBiLSTM model. The structures of LDA and Mask R-CNN are detailed in SMDCnet [15], while the ConvBiLSTM model is illustrated in [14]. This paper offers a detailed explanation of the enhancements applied to BiLSTM within ConvBiLSTM for crime prediction. Figure 2 depicts the outline of the suggested model.

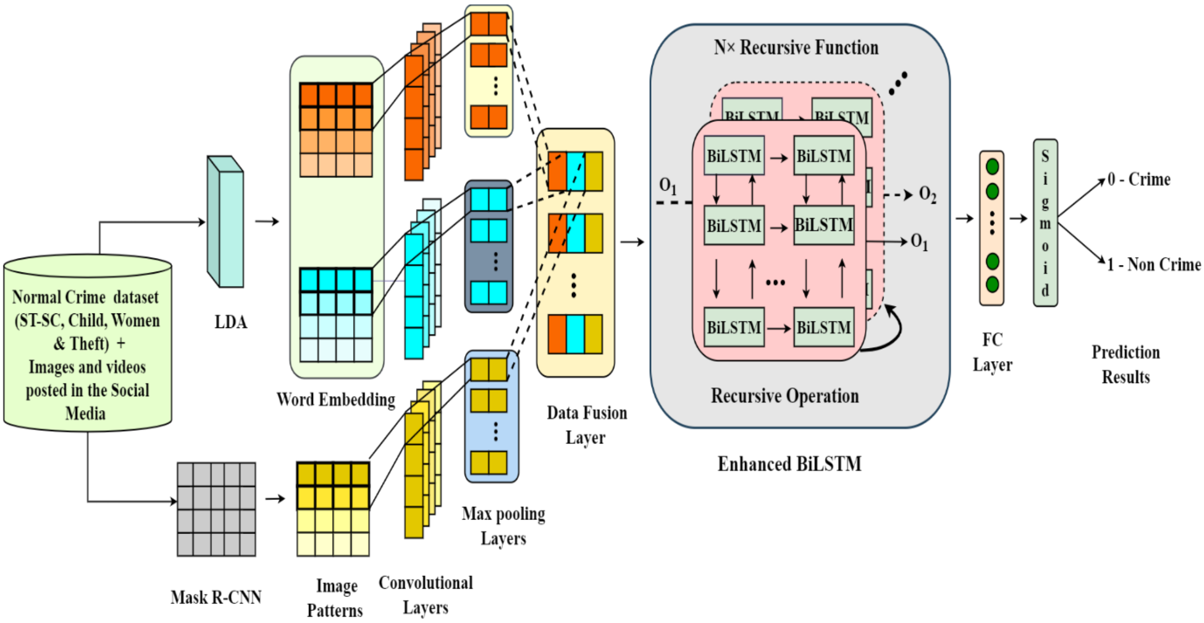

Figure 2 Complete Framework of the Suggested Methodology

3.3 Enhanced Recursive Self-Attention Mechanisms based BiLSTM

In ERSAM, Multi-Scale Self-Attention (MSSA) and recursive operations are used to eliminate information redundancy and prevent gradient explosion in BiLSTM model for increasing models accuracy.

3.3.1 Recursive Function

The initial recursive elements for forecasting crime detection modality employs a repeated weight sharing on organized input that enables the retrieval of complex hierarchical depictions. The recursive networks constitute the structured data and categorical components, primarily applied for compositional embedding in Natural Language Processing (NLP) tasks. A series of image patch embeddings serves as an input with no supplementary data utilized during the recursive cycles. To simplify recursive operations, consider employing two loops for network creation, as shown in Eq. (1),

$$\boldsymbol{R_l} = Q_l(Q_{l-1}(R_{l-1})) \quad (1)$$

Where, the recursive operation involves applying a function Q_1 to the result of another function Q_{l-1}. This processed the preceding layer R_{l-1}. This is visualized as a function Q_{l-1} applied initially and comes Q_l. For two recursive layers, the output R_l at layer l is calculated by passing the function output R_{l-1} through the function R_l.

3.3.2 Non – Linear Projection Layer (NLL)

NLL is used in dual recursive operations to perform sophisticated transformations between input and output modules. This minimizes unnecessary depictions learning by requiring dissimilarity between neighboring inputs and outputs. Eq. (2) provides the NLL formulation.

$$\boldsymbol{NLL}\,(\boldsymbol{R_{l-1}}) = \boldsymbol{MLP}\,(\boldsymbol{LN}(\boldsymbol{R'_{l-1}})) + \boldsymbol{R'_{l-1}} \quad (2)$$

In Eq. (9), MLP operates as a stacked transformation network which is equivalent to Feed-Forward Network (FFN), but provides different MLP proportions for unknown attributes. R'_{l-1} will be co-efficient of R_{l-1} with layer index.

3.3.3 Recursive Transformer Encoder

Eq. (3) defines the recursive transformer structure with dual iterative cycles within each blocks.

$$\boldsymbol{R_l} = \boldsymbol{NLL_2}\left(Q_{l-1}\left(\boldsymbol{NLL_1}(Q_{l-1}(\boldsymbol{R_{l-1}}))\right)\right) \quad (3)$$

where $\boldsymbol{R_{l-1}}$ and $\boldsymbol{R_l}$ denotes the input and output of each recursive operations accordingly. $\boldsymbol{NLL_1}$ and $\boldsymbol{NLL_2}$ applies the distinct and independent weights differing from MSSA and FFN that maintains uniform variable transforming throughout all the recursive functions within every blocks.

3.3.3 Recursive All-MLP

Recursive All-MLP [27] is defined in Eq. (4), Eq. (5), Eq. (6) which is illustrated below.

$$A_{*,x} = U_{*,x} + w_2 * GELU\,(w_1 * LN(U)_{*,x}) \quad (4)$$

$$V_{y,*} = A_{y,*} + w_4 * GELU\,(w_3 * LN(A)_{y,*}) \quad (5)$$

$$B_{y,*} = \mathcal{M}_{l-1}\left(\mathcal{M}_{l-1}(U_{*,x})\right) \quad (6)$$

Where, $A_{*,x}$ and $V_{y,*}$ represents the token-mixing and channel-mixing task. $A_{*,x}$ represents the the input tensor or feature map at a specific position $(*, x)$. GELU is the Gaussian Error Linear Unit applied to introduce non-linearity. w_1, w_2, w_3 and w_4 are the weights that are linearly transformed. $B_{y,*}$ defines the deeper transformation. $\mathcal{M}_{l-1}$ represents the MLP block, U and K represents the implicit dimensional space and the amount of number of non-adjacent image patches.

3.3.5 Gradients in a recursive block

Recursive transformers, unlike Deep Equilibrium Models (DEQs) [28] are not limited to finding input-output equilibrium in recursions. Direct backpropagation is implemented using precise operations in the ahead flow similar to the gradient descent approach, while constantly maintaining the hierarchy level and network estimations under regulation resulting in a minimal number of repetitive cycles. Typically, the weight derivative within every iterative module is represented in Eq. (7),

$$\frac{\partial S}{\partial W_Q} = \frac{\partial S}{\partial R^n}\frac{\partial R^n}{\partial W_Q} + \frac{\partial S}{\partial R^n}\frac{\partial R^n}{\partial R^{n-1}}\frac{\partial R^{n-1}}{\partial W_Q} + \cdots \frac{\partial S}{\partial R^n}\frac{\partial R^n}{\partial R^{n-1}} \cdots \frac{\partial R^n}{\partial R^n}\frac{\partial R^n}{\partial W_Q} = \sum_{x=1}^{n}\frac{\partial S}{\partial R^n}\left(\prod_{y=1}^{n-1}\frac{\partial R^{y+1}}{\partial R^y}\right)\frac{\partial R^x}{\partial W_Q}$$

(7)

In Eq. (7), W_Q represents a weight matrix associated with a particular layer in the network. S represents the objective operation. ∂W_Q is the gradient with respect to parameter with respect to W_Q. $\frac{\partial S}{\partial R^n} \cdot \frac{\partial R^n}{\partial W_Q}$ represents the direct contribution of the output R^n layer to the slope of the objective function with respect to W_Q. Similarly, $\frac{\partial S}{\partial R^n}\frac{\partial R^n}{\partial R^{n-1}}\frac{\partial R^{n-1}}{\partial W_Q}$ will be straight gradient of R^n and R^{n-1} will be respect to $R^{(n-1)}$ and the ∂W_Q. $\frac{\partial R^{y+1}}{\partial R^y}$ represents the gradient propagation from one layer to the next. $\prod_{y=1}^{n-1}\frac{\partial R^{y+1}}{\partial R^y}$ aggregates all the intermediate gradients.

3.3.6 Learnable Residual Connection (LRC)

Various techniques for short route paths in CNN are tested, demonstrating that the real residual setup with pre-action functions produces the best results. This model incorporates customizable scaling factors into each segment of the residual interactions, increasing the efficiency of BiLSTM model. The corresponding derivative is provided in Eq. (8), Eq. (9) and Eq. (10)

$$R_l' = \delta \times MSSA\ (LN(R_{l-1}) + \alpha \times R_{l-1})$$

(8)

$$R_l = \beta \times FFN\ (LN(R_l') + \gamma \times R_l')$$

(9)

$$NLL_{(R_{l-1})} = \omega \times MLP\ (LN(R_{l-1}') + \theta \times R_{l-1}')$$

(10)

where δ, $\alpha, \beta, \gamma, \omega, \theta$ are the learnable coefficients. R_l' is the integrative substitute of R_l. These variables are changed to 1 when the model parameters are simultaneously optimized without any enforced limitations.

3.5.7 Approximating MSSA for cost reduction

The recursive operation provides better representation by utilizing the same number of parameters but significantly increases training and inference overhead. To address this, diversed class self-attentions is introduced to effectively reduces Floating Point Operations Per Seconds (FLOPs) without compromising accuracy, despite the extra computational cost caused by recursion. The figure 3 portrays the approximating Global MSSA (G-MSSA) via MSSA with permutation.

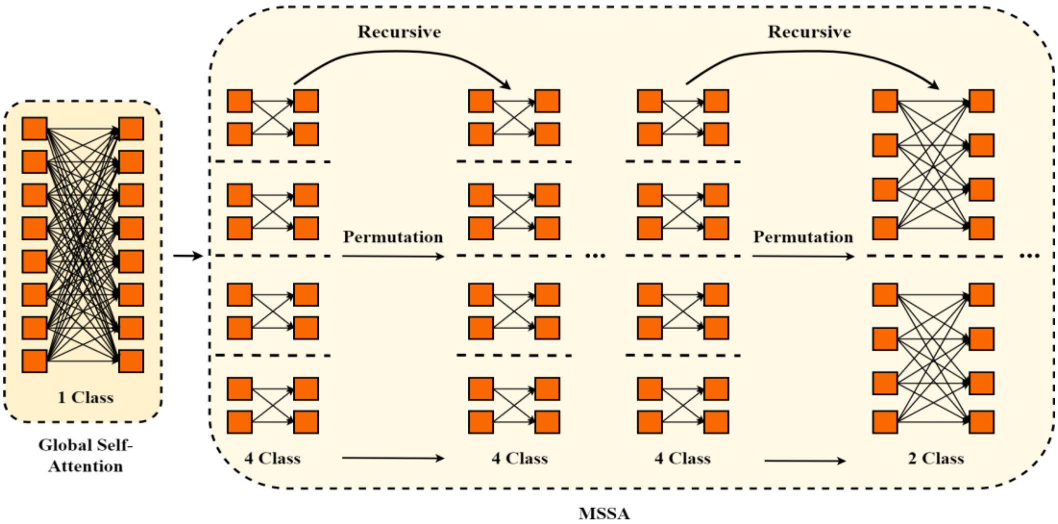

Figure 3 Estimating G-MSSA using MSSA with Permutation

Using MSSA in a recursive fashion, a self-attention module may be detached with the same or less computational cost. Depending on the trade-off between accuracy and FLOPs, the amount of categories in separate recursions may change. While the number of parameters remains the same, this method offers lower FLOPs and somewhat worse performance. Applying suitable self-attention segmentation allows the separation approach to achieve equivalent efficiency with fewer FLOPs. Using a BiLSTM structure, overall complexity is analyzed and group self-attentions across different group sizes are segmented for improving the models performances.

3.5.8 Gradients Accumulation Recursive MSSA

In approximating MSSA, the enhanced gradients accumulation is also to be considered. Assume $g_t = \nabla_\theta f_t(\theta)$ represents the gradient and then adam optimizer to optimize the naïve parameter update as $\theta_t \leftarrow \theta_{t-1} - \delta . \hat{e}_t^x / \left(\sqrt{\hat{b}_t^x} + \epsilon\right)$ where the gradients are inclined with respect to stochastic objective at timestep t is $z_t = \nabla_\theta f_t(\theta_{t-1})$. In this case, the first and second will be removed. After recursion, NLL guarantees $\hat{e}_t^x, \hat{b}_t^x$ discrepancy to provide new updating formula as $\theta_t \leftarrow \theta_{t-1} - \sum_{x=1}^{n} \delta . \hat{e}_t^x / \left(\sqrt{\hat{b}_t^x} + \epsilon\right)$ in which n is the iterative cycles. The learnt weights are more in line with the loss function and efficiency is automatically improved since recursion allows more updating/tuning of parameters in the same iteration. The recursion allows for more parameter updates/tuning in the same iteration, resulting in more aligned learnt weights with the loss function and improved efficiency.

3.4 Loss Function using CFIE

In the typical loss function of ConvBiLSTM, Mean Squared Error (MSE) is commonly used, but has significant limitations when dealing with insufficient data or when rapid model fitting is needed. MSE inherent structure can make optimization more challenging, particularly because it is not strictly convex in complex scenarios, leading to potential difficulties in reaching the global minimum during training. This non-convexity often requires a larger amount of data and more iterations to achieve optimal model performance. As a result, when data is limited, MSE struggles to facilitate quick and efficient convergence, which can prolong training times and reduce model accuracy.

To address these challenges, CFIE is proposed as the of MSE in the loss function. CFIE ensures stable optimization by providing a single global minimum, leading to smoother and more reliable gradient behavior. This helps prevent issues like gradient explosion or vanishing that often occur with deeper models like ConvBiLSTM. Furthermore, CFIE allows the model to fit the data more efficiently, even in scenarios where data is sparse, by improving convergence rates and optimizing the model with fewer iterations. Applying CFIE speeds up training, keeps performance stable, and improves the model's generalizability with less data. Eq. (11) represents the Information entropy expression.

$$H = E[-\,log\, p_x] = -\sum_{x=1}^{\delta} p_x\, log\, p_x \quad (11)$$

In Eq. (11), δ represents the window length.

The loss function of ConvBiLSTM using CFIE is given in Eq. (12)

$$Loss = -\sum_{x=1}^{J} \hat{p}_x \log \hat{p}_x - \left(-\sum_{x=1}^{\delta} p_x\, log\, p_x\right) \quad (12)$$

Where, $\hat{p} = \{\hat{p}_1, \hat{p}_2, \dots, \hat{p}_\delta\}$ represents the forecasting vector in which each $\hat{p}_x$ will be predicted value for x^{th} sample. $p = \{p_1, p_2, \dots, p_\delta\}$ defines the true observed vector in which each p is the actual value for x^{th} sample. In probabilistic models, both $\hat{p}_x$ and p vectors typically sum to 1. J is number of predicted samples in the forecasting vector $\hat{p}$. δ refers to the number of true observed samples in the true vector p. $log\, \hat{p}_x$ and $\log p_x$ are the logarithmic values of the predicted and true probabilities, respectively. The log function is applied across all the cross-entropy computations to evaluate the divergence among true distributions and predicted results.

3.6 Model Training

The hyperparameter configuration utilized to train the proposed model is listed in table 1.

Table 1. Parameter Settings

Parameters	Range
Word Embedding Size (Word2Vec)	300
Kernel Size	7
Filter Size	96
Pool Size	3
No. of BiLSTM layers	3
BiLSTM Output Size	128
Kernel Normalization	L_2 (0.001)

Dropout	0.5
Activation Operation	ReLU
Momentum	0.7
Batch size	124
Training rate	0.01
Scale Size	$5\ (1,3,\frac{n}{16},\frac{n}{8},\frac{n}{4})$
Length	4
Optimizer	Adam
Epochs	120
Loss function	Cross Entropy

Hence, the proposed model effectively eliminates gradient explosion in Bi-LSTM models due to redundant information that amplifies gradients. Eliminating this redundancy stabilizes training and enhances efficiency, leading to better learning and improved crime prediction performance. Below Algorithm effectively illustrates the SMDDCnet framework for crime prediction.

Algorithm: SMDDCnet for Crime Prediction
Input: Collected Dataset
Output: Eliminating BiLSTM limitations for crime prediction
1. **Begin**
2. Load the collected multimodal data including recorded crime data with images and videos from Twitter related to crime.
3. **\\Data Preparation**
4. Pre-process and normalize the multimodal data to eliminate noise and ensure consistency throughout model training.
5. **\\Feature Extraction**
6. Extract text vectors from crime data using Latent Dirichlet Allocation (LDA)
7. Apply Mask R-CNN with MFPN and NMS to retrieve image and video features from tweet data.
8. **\\ Eliminating BiLSTM issue**
9. Apply ERSAM to address the information redundancy issues in BiLSTM
10. Incorporate recursive compression to minimize redundant information.
11. Derive MSSA with variable scales to focus on different feature regions.
12. Use approximation in MSSA across recursive layers to reduce computational cost.
13. **\\ Prediction**
14. Apply ConvBiLSTM for crime prediction, utilizing CNN for local feature extraction and enhanced Bi-LSTM for bidirectional context representation of crime data.
15. **\\Loss Function**
16. Utilize CFIE as the loss function
17. Optimize the model parameters to minimize CFIE enhancing convergence speed and stability for crime prediction.
18. **End**

IV. RESULT AND DISCUSSION

This section evaluates the efficiency of the SMDDCnet model in Python 3.11, comparing it with existing models like AIST [18], SSO-BiLSTM [22] TL-BiLSTM [23], HDM-GNN [25] ConvBiLSTM [14] and SMDCnet [15]. The experiments were conducted on a system with Intel® CoreTM i5-4210 CPU @ 3GHz, 4GB RAM and a 1TB HDD running on Windows 10 64-bit. Both proposed and existing models were tested using datasets as described in Section 3.1. From the acquitted data, an overall sample of 9794 are determined where 7834 are used for training and 1960 are used for testing in split of 80:20 proportions. The confusion matrix for the suggested model is given in Figure 4.

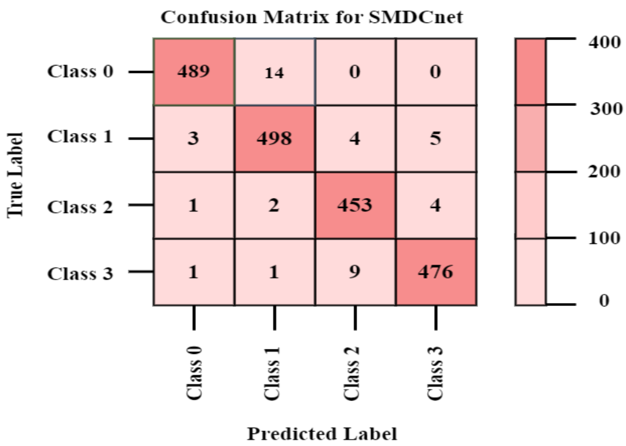

Figure 4 Confusion Matrix for the proposed model

The efficiency of the proposed model is evaluated using the following performance metrics.

- **Accuracy:** It is calculated as the ratio of correctly predicted instances to the total number of instances as given in Eq. (13)

$$Accuracy = \frac{TP + TN}{TP+TN+FP + FN} \quad (13)$$

Where, **True Positive (TP):** The system accurately identifies crime occurrences as a crime. **False Positive (FP):** The model wrongly detects a non-crime incident as a crime. **True Negative (TN):** The model accurately recognizes the non-crime event as a non-crime. **False Negative (FN):** The model wrongly predicts a crime event as a same event.

- **Precision:** The formulation is defines in Eq. (14)

$$Precision = \frac{TP}{TP+FP} \quad (14)$$

- **Recall:** The derivation is given in Eq. (15)

$$Recall = \frac{TP}{TP+FN} \quad (15)$$

- **F1-score:** It is the harmonic mean of precision and recall as provided in Eq. (16)

$$F1 - score = \frac{2\times Precision\times Recall}{Precision+Recall} \quad (16)$$

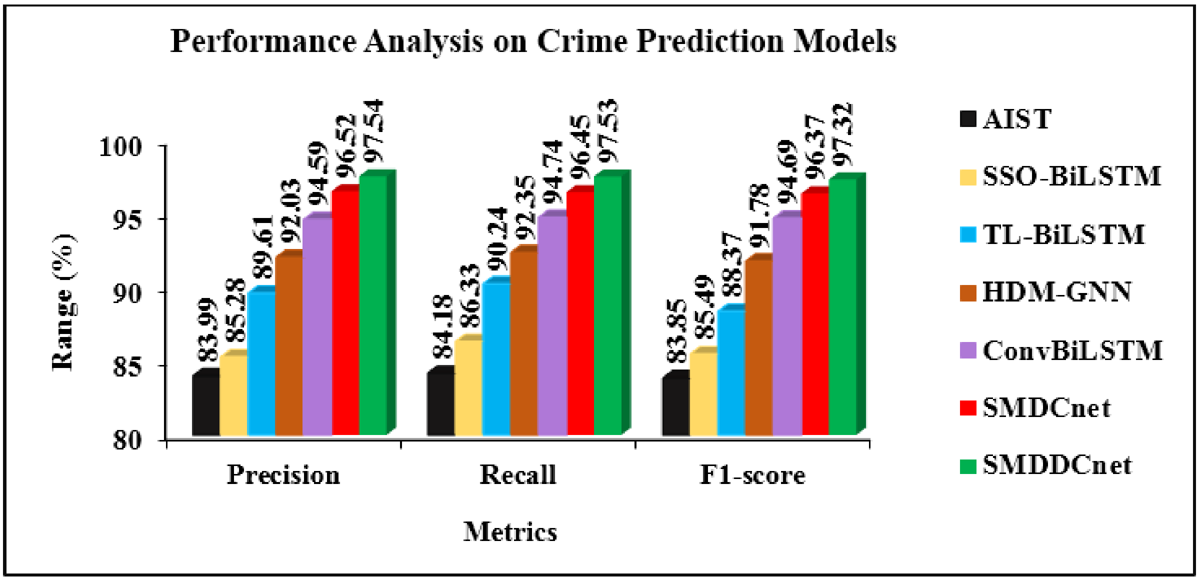

Figure 5 Performance Analysis of Different Crime Prediction Models on Collected Dataset

Figure 5 compares the performance of various crime prediction models using the Crime in India dataset. The SMDDCnet model outperforms others in precision, with increases of 16.24%, 14.48%, 8.95%, 6.08%, 3.21% and 1.15% over the AIST, SSO-BiLSTM, TL-BiLSTM, HDM-GNN, ConvBiLSTM and SMDCnet models, respectively. In terms of recall, SMDDCnet shows improvements of 16.13%, 14.38%, 8.85%, 5.99%, 3.12% and 1.06% compared to the same models. Additionally, the F1-score of SMDDCnet is higher by 15.86%, 12.97%, 8.08%, 5.61%, 2.94%, and 1.12% over the AIST, SSO-BiLSTM, TL-BiLSTM, HDM-GNN, ConvBiLSTM and SMDCnet models, respectively. These enhancements are due to improved contextual awareness which the proposed model captures the extended interactions in consecutive data such crime patterns or long temporal information.

Figure 6 illustrates the accuracy of various models evaluated on the crime data prediction dataset. The SMDDCnet model achieves accuracy that is 15.12% higher than AIST, 12.29% higher than SSO-BiLSTM, 10.34% higher than TL-BiLSTM, 7.1% higher than HDM-GNN, 2.87% higher than ConvBiLSTM, and 0.96% higher than SMDCnet. This improvement is attributed to SMDDCnet's enhanced representation learning and its focus on key inputs, which increases the accuracy by effectively capturing relationships between events, locations and times.

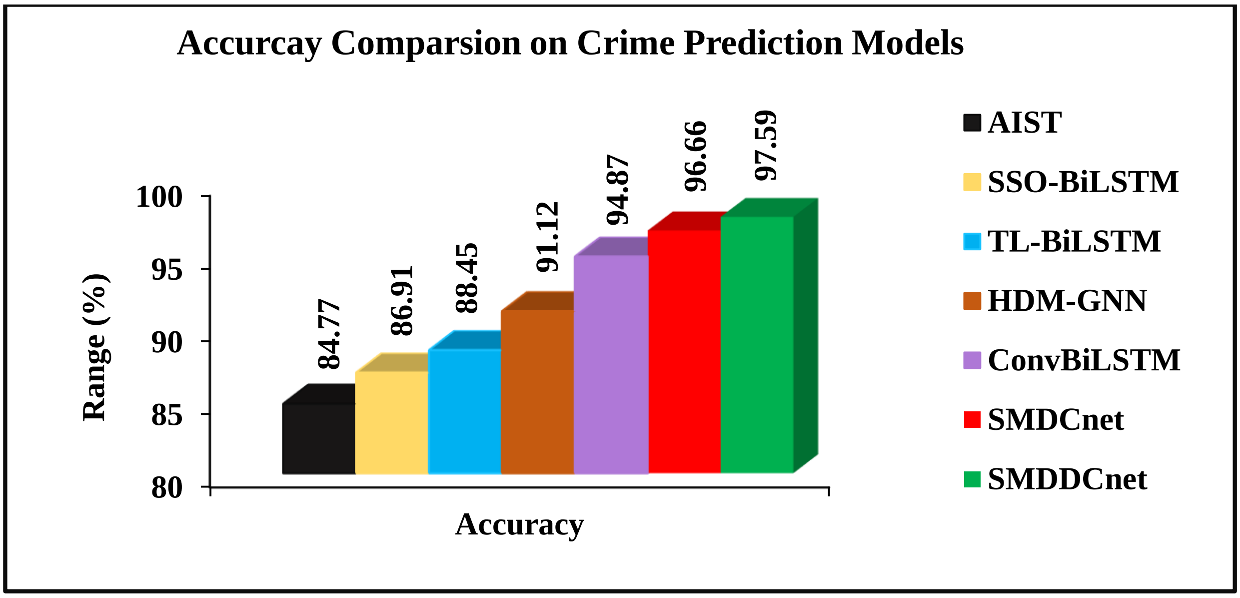

Figure 6 Accuracy Analysis of Different Crime Prediction Models on Collected Dataset

V. CONCLUSION

In this article, SMDDCnet is proposed to addresses issues in BiLSTM for crime prediction. This model involves two key enhancements: ERSAM and an advanced CFIE-based loss function. By processing data in bidirectional pathway, a bi-LSTM model with iterative operations minimizes feature dimensions oversees the sequences from both past and future contexts by extracting multi-level features via many stacked layers, MSSA in BiLSTM employs multi-scale attention heads to improve feature relationships, balancing broad context with local detail. An approximation method reduces cost while maintaining accuracy. CFIE in ConvBiLSTM improves optimization, prevents gradient explosions, and enhances crime prediction accuracy with limited data. Extensive experiments reveal that SMDDCnet achieves 97.54% accuracy on the Crime in India dataset, outperforming other crime prediction models.

REFERENCES

[1]. Sullivan, J. A. (2019). Achieving cumulative progress in understanding crime: Some insights from the philosophy of science. Psychology, Crime & Law, 25(6), 561-576.
[2]. DeKeseredy, W. S. (2015). Critical criminology. The handbook of deviance, 237-258.
[3]. Michalowski, R. J. (2016). What is crime?. Critical Criminology, 24, 181-199.
[4]. Alves, L. G., Ribeiro, H. V., & Rodrigues, F. A. (2018). Crime prediction through urban metrics and statistical learning. Physica A: Statistical Mechanics and its Applications, 505, 435-443.
[5]. Shamsuddin, N. H. M., Ali, N. A., & Alwee, R. (2017, May). An overview on crime prediction methods. In 2017 6th ICT international student project conference (ICT-ISPC) (pp. 1-5). IEEE.
[6]. Gupta, N. (2020). Influence of Social Media and Growth of Cyber Crimes-A Study. Issue 6, International Journal of Law Management & Humanities., 3, 861.
[7]. Prieto Curiel, R., Cresci, S., Muntean, C. I., & Bishop, S. R. (2020). Crime and its fear in social media. Palgrave Communications, 6(1).
[8]. Rodríguez-Ibánez, M., Casánez-Ventura, A., Castejón-Mateos, F., & Cuenca-Jiménez, P. M. (2023). A review on sentiment analysis from social media platforms. Expert Systems with Applications, 223, 119862.
[9]. Mahajan, R., & Mansotra, V. (2021). Correlating crime and social media: using semantic sentiment analysis. International Journal of Advanced Computer Science and Applications, 12(3).
[10]. Chen, X., Cho, Y., & Jang, S. Y. (2015, April). Crime prediction using Twitter sentiment and weather. In 2015 systems and information engineering design symposium (pp. 63-68). IEEE.
[11]. Stalidis, P., Semertzidis, T., & Daras, P. (2021). Examining deep learning architectures for crime classification and prediction. Forecasting, 3(4), 741-762.
[12]. Ateş, E. C., Bostancı, G. E., & Msg, S. (2020). Big data, data mining, machine learning, and

deep learning concepts in crime data. Journal of Penal Law and Criminology, 8(2), 293-319.

[13]. Kang, H. W., & Kang, H. B. (2017). Prediction of crime occurrence from multi-modal data using deep learning. PloS one, 12(4), e0176244.

[14]. Tam, S., & ÖzgürTanrıöver, Ö. (2023). Multimodal Deep Learning Crime Prediction Using Crime and Tweets. IEEE Access.

[15]. Sundari, P., & Malathi, A. (2025). Text and Visual Information Enriched Multimodal Deep Learning Model for Crime Prediction from Social media. International Journal of Intelligent Engineering and Systems, 18(2), 145-159.

[16]. Zhou, B., Chen, L., Zhao, S., Li, S., Zheng, Z., & Pan, G. (2022). Unsupervised domain adaptation for crime risk prediction across cities. IEEE Transactions on Computational Social Systems, 10(6), 3217-3227.

[17]. Vivek, M., & Prathap, B. R. (2023). Spatio-temporal Crime Analysis and Forecasting on Twitter Data Using Machine Learning Algorithms. SN Computer Science, 4(4), 383.

[18]. Rayhan, Y., & Hashem, T. (2023). AIST: An interpretable attention-based deep learning model for crime prediction. ACM Transactions on Spatial Algorithms and Systems, 9(2), 1-31.

[19]. Singuluri, P. K., Jayamma, B., Bharathi, B. D., Sree, T. B., & Santhoshi, B. L. (2023, June). Crime Prediction Using Modified Capsule Network with CrissCross Optimization on the Sentiment Analysis for Cyber-Security. In International Conference on Mining Intelligence and Knowledge Exploration (pp. 105-116). Cham: Springer Nature Switzerland.

[20]. Escobar, H., Cuevas, E., Toski, M., Ceron, F., & Perez-Cisneros, M. (2023). An agent-based model for public security strategies by predicting crime patterns. IEEE Access, 11(2023), 67070 – 67085.

[21]. Hashi, A. O., Abdirahman, A. A., Elmi, M. A., & Rodriguez, O. E. R. (2023). Deep Learning Models for Crime Intention Detection Using Object Detection. International Journal of Advanced Computer Science and Applications, 14(4).

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

[23]. Butt, U. M., Letchmunan, S., Hassan, F. H., & Koh, T. W. (2024). Leveraging transfer learning with deep learning for crime prediction. Plos one, 19(4), e0296486.

[24]. Zhou, B., Zhou, H., Wang, W., Chen, L., Ma, J., & Zheng, Z. (2024). HDM-GNN: A Heterogeneous Dynamic Multi-view Graph Neural Network for Crime Prediction. ACM Transactions on Sensor Networks.

[25]. Selvan, A. K., & Sivakumaran, N. (2024). Crime detection and crime hot spot prediction using the BI-LSTM deep learning model. International Journal of Knowledge-Based Development, 14(1), 57-86.

[26]. https://www.kaggle.com/datasets/rajanand/crime-in-india

[27]. Tolstikhin, I. O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., ... & Dosovitskiy, A. (2021). Mlp-mixer: An all-mlp architecture for vision. Advances in neural information processing systems, 34, 24261-24272.

[28]. Bai, S., Kolter, J. Z., & Koltun, V. (2019). Deep equilibrium models. Advances in neural information processing systems, 32.