A Deep Learning Algorithm to Cybersecurity: Enhancing Intrusion Detection with a Hybrid GRU and BiLSTM Model

Ameer A. Ghani
Information Networks Department, College of Information Technology, Babylon University, Babil, Iraq
ameerahmedg.net@student.uobabylon.edu.iq (corresponding author)

Suad A. Alasadi
Information Networks Department, College of Information Technology, Babylon University, Babil, Iraq
suad.alasady@uobabylon.edu.iq

Received: 21 February 2025 | Revised: 1 April 2025, 7 April 2025, and 9 April 2025 | Accepted: 12 April 2025

Licensed under a CC-BY 4.0 license | Copyright (c) by the authors | *DOI: https://doi.org/10.48084/etasr.10666*

ABSTRACT

Cyber security in networks and Internet of Things (IoT) environments is becoming complex with the evolution of sophisticated cyberattacks, and the existence of effective Intrusion Detection Systems (IDSs) is necessary. This work proposes a Network-based Intrusion Detection System (NIDS) for a hybrid Deep Learning (DL) model with Gated Recurrent Units (GRU) and Bidirectional Long Short-Term Memory (BiLSTM) to improve attack detection and classification. Pre-processing of datasets, feature selection with Pearson Correlation Coefficient (PCC), and training-testing with two benchmark datasets, CSE-CIC-IDS2018 and ToN_IoT, were performed. Surpassing standalone GRU and Bidirectional Long Short-Term Memory (BiLSTM) systems, the proposed hybrid model detected 99.86% of the attacks of the ToN_IoT dataset and 98.69% of the CSE-CIC-IDS2018 dataset while maintaining high accuracy, recall, and F1 score of over 99%. These results confirm that the proposed model can effectively counter traditional NIDS weaknesses through accuracy improvement in detections and with diversity and dynamics in networks' complex trends and IoT environments.

Keywords-cyber security; IoT; IDS; DL; GRU; BiLSTM

I. INTRODUCTION

Antivirus software, firewalls, and Intrusion Detection Systems (IDSs) are the most crucial techniques in cyber security. They protect networks from inner and outer attacks. An IDS is a detection system that protects the system through software and hardware inspections in a network [1]. As a result of the vast and rapid increase in cyber-attacks on IoT systems, people and companies have faced a wide range of issues related to credibility, enforcement, financing, and business operations [2]. Cyberattacks have generated critical security concerns, leading to the generation of new, flexible, and dependable IDSs. An IDS is a proactive intrusion detection tool utilized for on time intrusion and attack detection and security policy detection at the network and host level infrastructure. Intrusion detection, which is behavior-dependent, is categorized into Network-based Intrusion Detection Systems (NIDSs) and Host-based Intrusion Detection Systems (HIDSs) [3]. An NIDS monitors network traffic to spot anything unusual that might signal a cyber threat. Detecting changes from regular activity helps responding quickly to possible attacks [4]. A HIDS runs on a single computer or server and monitors its activity [5].

Machine Learning (ML) is a subsection of Artificial Intelligence (AI) that allows computer systems to do specific tasks independently. An ML system is built on learning from data to enable models to find trends, make predictions, and adapt decision-making [6]. There are three main learning types: Unsupervised, supervised, and reinforcement learning. In unsupervised learning, the model does not start with predefined groups. Instead, it identifies patterns and groups data based on similarities or statistical properties. The input data are not labeled, and the model figures out how to organize them independently. In supervised learning (classification), the categories are already known. The model is trained with labeled data, meaning it starts with input-output pairs and learns to classify new data [7], and reinforcement learning learns to make decisions based on prizes or punishments [6]. Deep Learning (DL) is a subset of AI and ML, originated in Artificial Neural Networks (ANNs). It has been applied to many research areas, including healthcare, visual recognition, text analytics, cybersecurity, etc. [8].

The increased number of cyber threats can often be noticed in internet traffic patterns [9]. There have been many trends in IDS employing DL and ML techniques, and part of such relevant works will be discussed below.

Authors in [10] proposed a DL-based method for IDS improvement using the CSE-CICIDS2018 dataset with 14 classes of assault and 76 attributes. This work focuses mainly on the class imbalance problem by performing up-sampling and down-sampling, and the Convolutional Neural Network (CNN) model and Long Short-Term Memory (LSTM) model were used to optimize the performance. The highest score of CNN on accuracy was 98.31%, whereas that of LSTM was 98.15%, while the latter reported a low loss of 0.0403%. Advantages comprised enhanced multi-class attack detection with efficient training time from CNN. While discussing their limitations, the authors stated that the prolonged training times on LSTM can hamper results. Authors in [11] proposed a Metaverse-IDS to detect intrusive activity in IoT networks used within the environment of Metaverse, using a DL technique. This included feature extraction by Kernel Principal Component Analysis (PCA) and using CNNs for classifying an attack. Two benchmark, Bot-IoT and ToN-IoT [23] were considered. The experimental performance achieved the best accuracy at 99.8% with less than 0.2 False Negative Rate. This approach provides better efficiency in attack detection with reduced computational overhead. However, the model was tested on benchmark datasets rather than real-world Metaverse-IoT environments, which may not be appropriate.

Authors in [12] proposed a hybrid DL-based IDS to enhance security in IoT networks by overcoming the limitations of single-layer IDS models. The authors used RNN and GRU to identify intrusions in all three layers of IoT architecture: perception, network, and application. It was trained and tested on ToN-IoT, a publicly available multilayer IoT security research dataset. The proposed system attained an accuracy of 99% on network flow data and 98% for overall model performance in application layer data, performing well compared to the general approaches based on ML and DL. Among its benefits, some advantages of the proposed model include the high accuracy value, improved detection of various attacks, and real-time intrusion detection. However, this model was based on the ToN-IoT dataset and may not generalize in other attack scenarios; therefore, validation on more diversified datasets is necessary. Authors in [13] tackled the increasing threat of network intrusions within IoT applications by designing a BiLSTM-CNN Hybrid IDS. The approach used BiLSTM to learn the time relationships and CNN to learn the spatial patterns of the features. With the UNSW-NB15 dataset, the hybrid approach outperformed both standalone BiLSTM and CNN models with a 99.265% precision and 97.51% accuracy. The research showcases the model's strength in learning the attacks with fewer false alarms while increasing the classification performance. The strengths of the approach are the high precision of the detections, the flexibility to learn the behavior of the IoT networks and the ease of computability both on the GPU and the CPU. Its weaknesses are the possible overfitting with increased training time and the increased need for computational resources. Future research is recommended to improve the approach to multi-class classification.

Authors in [14] addressed the challenge of IDS in network security by employing ML-based classification methods on the CSE-CIC-IDS2018 dataset. The prepared data set compared six classifiers in classification performance: Random Forest, Gradient Boosting, XGBoost, CatBoost, Logistic Regression, and LightGBM. The accuracy of all the models was high, while for XGBoost, LightGBM, and CatBoost, it reached 98%. The result gives a clear point based on ensemble-based classifiers that could be effective for intrusion detection in networks. The proposed scheme outperformed previous techniques in detectability. However, a single dataset alone was considered in the work, so the generalization issue remains. Authors in [15] proposed an IDS with DL techniques for IoT networks. The mechanism is designed to improve several attack detections with reduced computational complexity. The ToN-IoT dataset was considered. The paper presents four types of training models: LSTM, Bi-LSTM, GRU, and a self-attention mechanism with GRU. Hyperparameters were optimized using the Grid Search algorithm to search for an apt learning rate and hidden units. The best detection accuracy (99%) was achieved with the model with a self-attention mechanism and with the GRU model (97% and 98.1% with LSTM and 98.4% with Bi-LSTM). Minimizing this model reduced classification time by up to 84% compared to GRU, improving efficiency for real-time detection. The model presented a better balance of accuracy and efficiency, although it is interesting to consider applying more advanced techniques for DL in future work.

Authors in [16] studied the efficiency of feature extraction techniques and ML models to improve IDS for IoT networks. They employed three feature extraction methods: PCA, Linear Discriminant Analysis (LDA), and Auto-Encoders (AE), along with deep and shallow learning models like CNN, DFF, and Decision Trees for performance evaluation on the UNSW-NB15, ToN-IoT, and CSE-CIC-IDS2018 datasets. The results showed that PCA and AE performed better compared to LDA. The highest classification accuracy was 98.67% by CNN on UNSW-NB15 with AE. The most essential advantages included a reduction in the dimensionality of the data, as well as the improvement of the attack detection rate. However, no combination is optimal for every dataset since performance varies seriously because of features in different datasets. Authors in [17] compared feature selection against feature extraction methods for optimizing an IDS in IoT environments. They proposed a three-stage ML framework using the ToN-IoT dataset: pre-processing, feature reduction, and classification. The results indicated that compared with the feature selection scheme, PCA of feature extraction resulted in high performance, reaching 86.83% detection. On the other hand, the computational cost of training and testing using the PCA technique was comparatively superior to that obtained through the method based on feature selections.

The limitations identified from related works provide a background. This research proposes a hybrid DL for handling security challenges while considering known datasets CSE-CIC-IDS2018 and ToN_IoT. The proposed hybrid DL algorithm, consisting of the GRU and BILSTM models, improves detection accuracy while making the systems more adaptive to complex and continuously evolving environments. The hybrid model detects patterns of dependencies in network traffic data well and yields robust results in finding malicious activities that enhance the security of the general system.

II. METHODOLOGY

The proposed methodology can be seen in Figure 1.

A. Data Collection and Pre-processing.

Two datasets, CSE-CIC-IDS2018 [18] and ToN-IoT [19] were utilized in the proposed work.

1) CSE-CIC-IDS2018

CSE-CIC-IDS2018 [18] is a cybersecurity dataset for IDS training. It includes labeled network traffic data representing normal activities and various attacks captured in realistic environments using 50 attack machines and a victim organization comprising five departments, 420 machines, and 30 servers. The dataset has captured network traffic and system logs. Overall, the dataset contains 80 features extracted by CICFlowMeter-V3 [20].

2) ToN_IoT Dataset

The ToN_IoT (Telemetry and Network IoT) benchmark dataset is a rich benchmark for IDS testing in IoT, network, and telemetry settings [19]. It integrates various sources of information, such as network traffic, IoT device and sensor telemetry, and OS logs, and simulates real IoT settings under both attack and baseline settings. Its network flow information has 44 features, such as a combination of statistics (e.g. packet count, flow duration) and network-related features (e.g. protocol, source and target IPs, and ports). It is broadly utilized for cybersecurity studies, particularly for detection of attacks such as DoS, DDoS, reconnaissance, and data exfiltration in IoT networks [21-28].

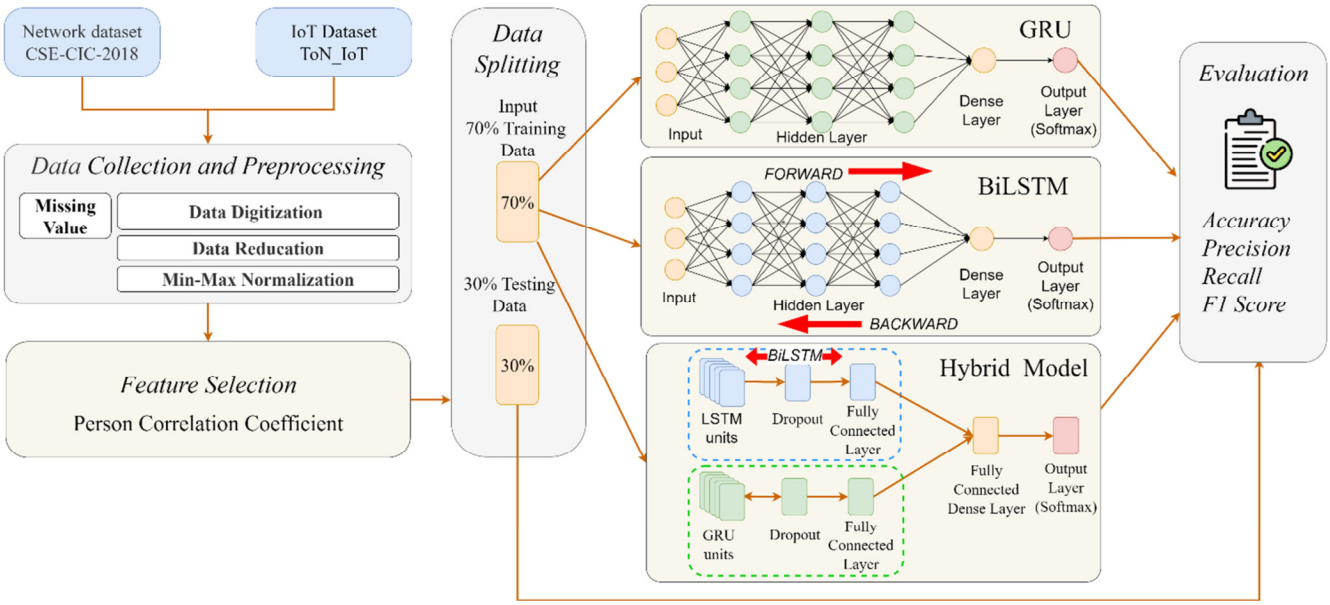

Fig. 1. Proposed system architecture.

3) Pre-processing

Raw data were preprocessed to enhance their suitability for DL models.

- The average mean is used to calculate the null value and label encoding is used that converts records categorical ones such as (tcp and udp ……, etc.) into numerical values. Operations are involved in converting each textual value into an integer value concerning its position in a sequence [29]. If the unique value is set in a field with n records, say A = {a0, a1, a2, . . ., ak−1}, then the form substitution in text data happening in this field is made by:

$$\mathrm{TEnc(bi)} = \begin{cases} 0, \mathrm{bi} = \mathrm{a0}; \\ 1, \mathrm{bi} = \mathrm{a1}; \\ 2, \mathrm{bi} = \mathrm{a2}; \\ \quad . \\ \quad . \\ \quad . \\ \mathrm{k}-1, \mathrm{bi} = \mathrm{ak}-1; \end{cases} \quad (1)$$

- Data Reduction with Information Entropy: By grouping records simultaneously to minimize the dataset size, statistical patterns and correlations in network traffic can be captured efficiently, reducing overall data size while preserving meaningful information. Entropy calculated the randomness of these packet groups. This probability-based approach helps identify structured patterns [30], as shown in (2). In this work, we tested data reduction-based entropy on 25, 50, and 75 and found that the 25 grouping records achieved the best result.

$$\mathrm{H(X)} = -\sum_{i=1}^{n} \mathrm{p(xi)log2p(xi)} \quad (2)$$

- Min-Max Normalization: Scaling the range of features to a uniform range from 0 to 1, reduces the impact of high-magnitude features (3):

$$\mathrm{v'} = \frac{\mathrm{v}-\mathrm{min_A}}{\mathrm{max_A}-\mathrm{min_A}}(\mathrm{new\_max_A} - \mathrm{new\_min_A}) \quad (3)$$

B. Feature Selection

Following preprocessing, feature selection identifies the most important features of the Pearson Correlation Coefficient (PCC), which is a computational tool for approximating the linear relation between features, reducing computational complexity and improving model performance. PCC achieves the best results in feature selection because it directly quantifies the linear relationship between each feature and the target variable, ensuring that only the most relevant features are retained. Unlike chi-square, which does not measure correlation strength, and PCA as feature extraction, which transforms features into uncorrelated components without considering the target variable, PCC maintains interpretability while effectively identifying predictive features. Therefore, in this research, the PCC achieved the best feature selection result compared to chi-square and PCA. PCC is calculated by:

$$\text{corr}(X,Y) = \frac{\text{cov}(X,Y)}{\sigma x \sigma y} \quad (4)$$

Feature selection using PCC for the ToN_IoT and CSE-CIC-IDS2018 datasets consisted of 22 and features, respectively.

C. Data Splitting

After trials, we found that data division in 70% for training and 30% for testing achieved the best results:

D. Deep Learning Algorithms

1) Long Short-Term Memory (LSTM)

LSTM networks are well-suited for tasks where the order of data is important, as they can recognize and retain temporal dependencies in input data. These networks contain memory cells that allow for the retention of long-term information. Each memory cell includes multiple gates that regulate the flow of information and maintain the cell's state. Specifically, LSTM networks consist of three key components: input gates, output gates, and forget gates, which work together to manage the storage and processing of information efficiently [31]. The LSTM unit comprises three primary components, often referred to as gates: the forget gate, the input gate, and the output gate, as can be seen in Figure 2.

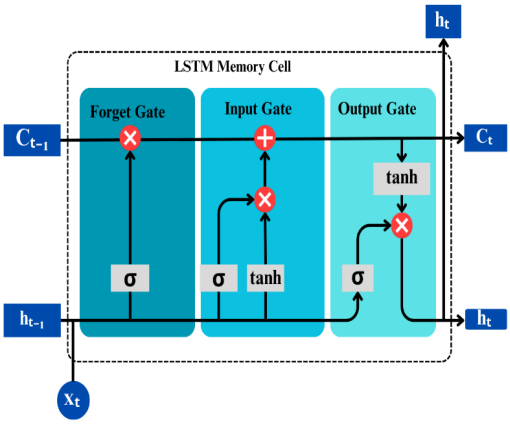

Fig. 2. LSTM architecture.

2) BiLSTM

The Bi-LSTM takes past and future information in both directions and can capture both. In contrast to a simple LSTM, whose input is one direction, BiLSTM permits information to flow through both reverse and forward layers. BiLSTM is applied in most of its types of operations, such as forecasting and text classification [32].

3) Gated Recurrent Units (GRU)

The GRU is a type of RNN with a simpler architecture than the LSTM network. While LSTM introduces three gates to regulate information flow, the GRU model has a more streamlined structure with one less gate. Despite this difference, GRU retains characteristics similar to LSTM and is often considered a more efficient alternative in specific applications [33]. GRU utilizes a less complex model structure for sequential information processing through gate incorporation, as in Figure 3. Gating aids in effectively processing long-term dependencies and keeps the model less complex than LSTMs.

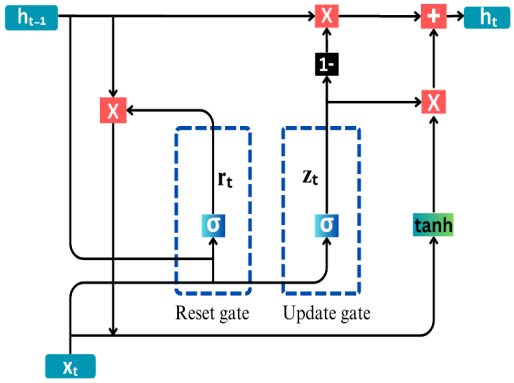

Fig. 3. GRU architecture.

4) Hybrid BiLSTM and GRU

The hybrid DL model utilizes GRU and BiLSTM networks to enhance the sequential processing capacities of information. Two branches make its structure: a GRU path and a BiLSTM path, both processing similar input information. In the GRU path, a sequence of several GRU layers follows one another, and with them, the model can easily extract temporal relations in sequential information. In the BiLSTM path, layers are utilized for processing data in both directions, enhancing contextual comprehension capacities in the model. The output of both branches is then merged with a concatenation layer and then with a fully dense connected layer from both models.

E. Classification

The classification stage utilizes trained DL models to sort information into two categories, attack (malicious activity) and normal (benign activity).

F. Evaluation

For the performance evaluation, accuracy, recall, precision, and F1 score metrics were utilized [34]. F1 score can be utilized in cases with an unbalanced distribution of datasets. These metrics are defined by:

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (5)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (6)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (7)$$

$$\text{F1 Score} = 2 \times \frac{\text{precision} \times \text{Recall}}{\text{precision+Recall}} \quad (8)$$

where TP, TN, FP, and FN represent True Positives, True Negatives, False Positives, and False Negatives, respectively.

III. RESULT AND DISCUSSION

The experiments were performed on a computer system with an AMD Ryzen 5 5600 CPU with 16 GB RAM and an 8 GB GDDR6 RAM with an NVIDIA GeForce RTX 2070 utilized for computation purposes. Performance evaluation of the proposed system scheme was performed through experiments, implementing the system in Python (PyCharm). CIC-CSE-2018 and ToN_IoT datasets were considered for representing network behavior and attack trends, both highly important for testing robustness and accuracy in NIDS. The proposed hybrid DL model outperforms the standalone models by combining the strengths of GRU and BiLSTM networks for enhanced sequential data processing. GRU efficiently captures long-term dependencies while mitigating the vanishing gradient problem, making them practical for extracting temporal relationships and BiLSTM processes data bidirectionally, leveraging past and future context for improved comprehension. The model integrates efficient sequence modeling with comprehensive contextual learning by merging outputs from both architectures. This fusion enhances feature extraction, robustness, and generalization, enabling superior performance in sequence-dependent tasks compared to the standalone models.

Table I presents the key hyperparameters employed models on the Ton_IoT dataset. The same hyperparameter and models were used on the CSE-CIC-IDS2018 dataset, except for the Output layer neurons, which consist of 14 attack types and one normal class.

TABLE I. HYPERPARAMETERS FOR GRU, BILSTM, AND HYBRID DL IN TON_IOT DATASET

Hyperparameter	**GRU**	**BiLSTM**	**Hybrid GRU +BiLSTM**	
Number of layers	4 (3 GRU + 1 Dense layer)	4 (3 BiLSTM + 1 Dense layer)	4 (3 GRU + 1 Dense layer), 4 (3 BiLSTM + 1 Dense layer)	
First layer neurons	128	128	128 GRU	128 BiLSTM
Second layer neurons	64	64	64 GRU	64 BiLSTM
Third layer neurons	32	32	32 GRU	32 BiLSTM
Number of neurons in the dense layer	128	128	128	128
Dropout	0.2	0.2	0.2	0.2
Output layer neurons	9 + normal class	9 + normal class	9 + normal class	
Hidden layer activation function	Relu	Relu	Two Relu	
Output layer activation function	Softmax	Softmax	Softmax	
Optimizer	Adam	Adam	Adam	
Learning rate	0.005	0.005	0.005	
Batch Size	128	128	128	
Number of epochs	50	50	50	

The confusion matrix illustrates a critical analysis of performance for all algorithms. Figures 4 and 5 illustrate the confusion matrices of the algorithms in both datasets. It can be seen that the proposed hybrid model outperforms the GRU and BiLSTM.

All three models perform well to detect positive cases, with the Hybrid GRU+BiLSTM model having the best performance. In ToN_IoT, it has the highest TP (261,552), the minimum FN (207), and the minimum FP (128), with better accuracy and reliability for the classification of the two traffic classes, i.e., the attack and benign traffic, compared with the individual GRU and BiLSTM models as shown in Figure 4. The best classification of the CIC-CSE-2018 dataset is produced by the Hybrid GRU+BiLSTM model with maximum TP (80,458) and TN (33,699) and minimum FP (174), with fairly low FN (1,374). Compared with the individual GRU and BiLSTM, the hybrid model is more accurate and better balanced for identifying the two traffic classes, as depicted in Figure 5.

Fig. 4. Confusion matrix for the ToN_IoT dataset.

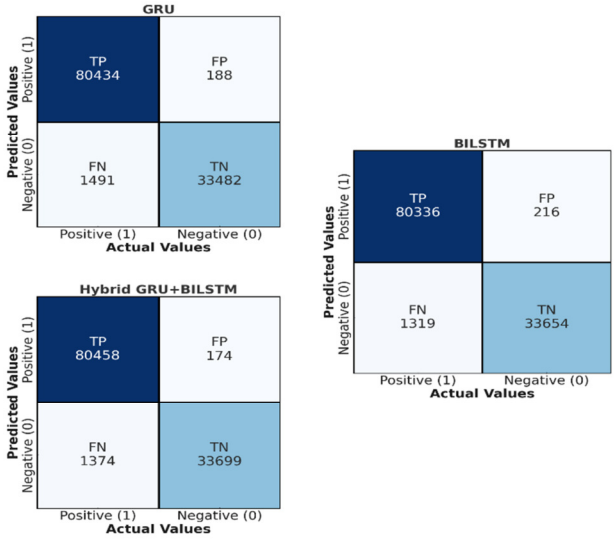

Fig. 5. Confusion matrix for the CSE-CIC-IDS2018 dataset.

A. Results of on the ToN_IoT Dataset

This section compares the performance of the ToN_IoT dataset of the considered models. Table II shows the results. The Hybrid GRU and BILSTM model had the best accuracy (0.9986), precision (0.9992), recall (0.9993), and F1 score (0.9992) and hence was the best-performing model.

B. Result of DL for CSE-CIC-IDS2018

CSE-CIC-IDS2018, a network traffic-related cybersecurity dataset, was used to test model performance in overcoming intrusion-related issues such as anomalous behavior and traffic analysis. CSE-CIC-IDS2018 performance, through its use with a range of datasets, validates the adaptability and effectiveness of the proposed system in processing a range of datasets and anomalous behavior in cybersecurity. Results on the CSE-CIC-IDS2018 are described in Table III. The Hybrid GRU+BILSTM model had the best accuracy (0.9869), precision (0.99426), recall (0.9985), and F1 score (0.9901). This illustrates that the hybrid model improves classification performance, with better intrusion detection compared to standalone GRU and BILSTM models.

TABLE II. RESULTS FOR TON_IOT DATASET

Model	Accuracy	Precision	Recall	F1 Score
BILSM	0.9984	0.9991	0.9993	0.9992
GRU	0.9980	0.9449	0.9738	0.9591
Hybrid GRU+ BILSTM	0.9986	0.9992	0.9993	0.9992

TABLE III. RESULTS FOR CSE-CIC-IDS2018 DATASET

Model	Accuracy	Precision	Recall	F1 Score
BILSM	0.9867	0.9936	0.9622	0.9777
GRU	0.9860	0.9818	0.9607	0.9772
Hybrid GRU+ BILSTM	0.9869	0.9942	0.9985	0.9901

Table IV shows the comparison results of the proposed algorithm with similar works, utilizing ToN_IoT and CSE-CIC-IDS2018 datasets.

TABLE IV. COMPARISON WITH THE PROPOSED WORK

Ref/Year	Dataset	Model	Accuracy	Precision	Recall	F1 Score
[11] /2023	BoT-IoT ToN_IoT	CNN	99.8% ToN_IoT	97.6% ToN_IoT	99.9% ToN_IoT	98.9% ToN_IoT
			99.8% Bot IoT	99.3% Bot IoT	99.9% Bot IoT	99.7% Bot IoT
[12] /2023	ToN_IoT	hybrid model RNN and GRU	99%	99%	98%	97%
[13] /2023	UNSW-NB15	CNN, BiLSTM, BiLSTM-CNN	87.15%	98.123%	82.294%	89.514%
			88.22%	98.167%	83.697%	90.356%
			91.14%	98.265%	87.641%	93.091%
[14] /2024	CSE-CIC-IDS2018	RF, gradient boosting, XGBoost, CatBoost, LR, LightGBM	98% LightGBM	98% LightGBM	100% LightGBM	99% LightGBM
[15] /2024	ToN_IoT	LSTM, BiLSTM, GRU, GRU-based self-attention mechanism	99% GRU-based self-attention mechanism	99% GRU-based self-attention mechanism	99% GRU-based self-attention mechanism	99% GRU-based self-attention mechanism
[16] /2024	UNSW-NB15, ToN_IoT, CSE-CIC-IDS2018	DFF, CNN, RNN, DT, LR, NB	98.23% DT with TON IoT	99% DT with TON IoT	98.28% DT with TON IoT	97% DT with TON IoT
			98.02% DT with CSE-CIC-2018	93% DT with CSE-CIC-2018	94.76% DT with CSE-CIC-2018	97% DT with CSE-CIC-2018
Proposed	ToN_IoT	Hybrid GRU and BiLSTM	99.86%	99.92%	99.93%	99.92%
	CSE-CIC-IDS2018	Hybrid GRU and BiLSTM	98.69%	99.42%	99.85%	99%

Our work had an improved and efficient preprocessing that helped the model perform better, be more accurate, and run faster. Similar research often overlooks these steps. Authors in [11, 12], used basic methods such as simple data normalization or balancing but did not mention reducing unnecessary data or choosing features based on their relationships.

In contrast, more detailed, multi-step preprocessing approaches, such as label encoding, average mean, and entropy-based data reduction, fare better. These steps speed up training and keep key signs of attacks, something not done in past studies, and Min-Max normalization was utilized to scale all features to the same range, which helps the model learn faster and better.

PCC picks the most useful and correlated features for feature selection. This approach keeps the features understandable while reducing data size and avoiding overfitting. After improved preprocessing and feature selection, the hybrid model learns from cleaner, more relevant, and smaller data, leading to excellent results, 99.86% accuracy on the ToN_IoT dataset, and 98.64% on CSE-CIC-IDS2018, surpassing the GRU-based self-attention mechanism of [15]. Combining preprocessing, feature selection, and a hybrid model with a strong GRU model for speed and BiLSTM to understand patterns in both directions helps the proposed system better detect attacks. Our contribution is thus a great leap forward for intelligent, adaptive IDSs.

IV. CONCLUSION

The research compares the performance of GRU, BiLSTM, and their hybrid model. It concludes that the hybrid model of GRU and BiLSTM can make NIDS more reliable and accurate, particularly in IoT environments' security. The hybrid model achieved high performance in the ToN_IoT and CSE-CIC-IDS2018 datasets, with 99.86% and 98.69% detection accuracy, respectively, confirming that a complex cybersecurity attack can be handled more effectively with a hybrid model than with a standalone GRU and BiLSTM model. Therefore, the proposed model can be considered adequate for performance evaluation.

The model is optimized and efficient, using techniques such as the PCC for feature selection and preprocessing. It also performs efficiently in various cyberattack types, such as DDoS, password, and ransomware. GRU and BiLSTM perform well when evaluated separately, but in the proposed hybrid model, accuracy, precision, recall, and F1 values increased with in various scenarios.

Future work could deal with further optimizing and expanding the hybrid DL model to new areas, paving the path for even broader use and expandability in cybersecurity.

REFERENCES

[1] H. Liu and B. Lang, "Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey," *Applied Sciences*, vol. 9, no. 20, Jan. 2019, Art. no. 4396, https://doi.org/10.3390/app9204396.

[2] C. Yin, Y. Zhu, J. Fei, and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017, https://doi.org/10.1109/ACCESS.2017.2762418.

[3] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," *IEEE Access*, vol. 7, pp. 41525–41550, 2019, https://doi.org/10.1109/ACCESS.2019.2895334.

[4] A. A. Abdullah and S. A. Hussein, "Detection and Mitigation Distribution Denial of Service Attack Based on Blockchain Concept," *Ingénierie des Systèmes d'Information*, vol. 29, no. 3, pp. 1043–1049, Jun. 2024, https://doi.org/10.18280/isi.290322.

[5] L. Ashiku and C. Dagli, "Network Intrusion Detection System using Deep Learning," *Procedia Computer Science*, vol. 185, pp. 239–247, Jan. 2021, https://doi.org/10.1016/j.procs.2021.05.025.

[6] B. R. Yadav, "Machine Learning Algorithms: Optimizing Efficiency in AI Applications," *International Journal of Engineering and Management Research*, vol. 14, no. 5, pp. 49–57, Jul. 2024, https://doi.org/10.5281/zenodo.14005017.

[7] E. F. Abdullah, A. A. Lafta, and S. A. Alasadi, "Information Gain-Based Enhanced Classification Techniques," in *Next Generation of Internet of Things*, 2021, pp. 499–511, https://doi.org/10.1007/978-981-16-0666-3_40.

[8] A. Fatani, M. Abd Elaziz, A. Dahou, M. A. A. Al-Qaness, and S. Lu, "IoT Intrusion Detection System Using Deep Learning and Enhanced Transient Search Optimization," *IEEE Access*, vol. 9, pp. 123448–123464, 2021, https://doi.org/10.1109/ACCESS.2021.3109081.

[9] S. A. Alasadi and W. S. Bhaya, "Anomaly Detection System for Internet Traffic based on TF-IDF and BFR Clustering Algorithms," *International Journal of Engineering and Technology*, vol. 7, no. 4.19, pp. 600–604, Nov. 2018, https://doi.org/10.14419/ijet.v7i4.19.27967.

[10] A. A. Hagar and B. W. Gawali, "Deep Learning for Improving Attack Detection System Using CSE-CICIDS2018," *NeuroQuantology*, vol. 20, no. 7, pp. 3064–3074, Aug. 2022.

[11] T. Gaber, J. B. Awotunde, M. Torky, S. A. Ajagbe, M. Hammoudeh, and W. Li, "Metaverse-IDS: Deep learning-based intrusion detection system for Metaverse-IoT networks," *Internet of Things*, vol. 24, Dec. 2023, Art. no. 100977, https://doi.org/10.1016/j.iot.2023.100977.

[12] N. W. Khan *et al.*, "A hybrid deep learning-based intrusion detection system for IoT networks," *Mathematical Biosciences and Engineering*, vol. 20, no. 8, pp. 13491–13520, 2023, https://doi.org/10.3934/mbe.2023602.

[13] S. Sadhwani, M. A. H. Khan, R. Muthalagu, and P. M. Pawar, "BiLSTM-CNN Hybrid Intrusion Detection System for IoT Application." Research Square, Jan. 03, 2024, https://doi.org/10.21203/rs.3.rs-3820775/v1.

[14] H. İ. Coşar, Ç. Arısoy, and H. Ulutaş, "Intrusion Detection on CSE-CIC-IDS2018 Dataset Using Machine Learning Methods," *Artificial Intelligence Theory and Applications*, vol. 4, no. 2, pp. 143–154, Oct. 2024.

[15] M. L. Mutleg, A. M. Mahmood, and M. M. J. Al-Nayar, "Deep Learning Based Intrusion Detection System of IoT Technology: Accuracy Versus Computational Complexity," *International Journal of Safety and Security Engineering*, vol. 14, no. 5, pp. 1547–1558, Oct. 2024, https://doi.org/10.18280/ijsse.140522.

[16] M. Sarhan, S. Layeghy, N. Moustafa, M. Gallagher, and M. Portmann, "Feature extraction for machine learning-based intrusion detection in IoT networks," *Digital Communications and Networks*, vol. 10, no. 1, pp. 205–216, Feb. 2024, https://doi.org/10.1016/j.dcan.2022.08.012.

[17] J. Li, M. S. Othman, H. Chen, and L. M. Yusuf, "Optimizing IoT intrusion detection system: feature selection versus feature extraction in machine learning," *Journal of Big Data*, vol. 11, no. 1, Feb. 2024, Art. no. 36, https://doi.org/10.1186/s40537-024-00892-y.

[18] Canadian Institute for Cybersecurity, "A Realistic Cyber Defense Dataset (CSE-CIC-IDS2018)." Available at: https://registry.opendata.aws/cse-cic-ids2018

[19] N. Moustafa, "The ToN_IoT Datasets," UNSW Canberra at ADFA, [Online]. Available: https://research.unsw.edu.au/projects/toniot-datasets.

[20] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," presented at the 4th International Conference on

Information Systems Security and Privacy, May 2025, pp. 108–116, https://doi.org/10.5220/0006639801080116.

[21] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," *Sustainable Cities and Society*, vol. 72, Sep. 2021, Art. no. 102994, https://doi.org/10.1016/j.scs.2021.102994.

[22] T. M. Booij, I. Chiscop, E. Meeuwissen, N. Moustafa, and F. T. H. den Hartog, "ToN_IoT: The Role of Heterogeneity and the Need for Standardization of Features and Attack Types in IoT Network Intrusion Data Sets," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 485–496, Jan. 2022, https://doi.org/10.1109/JIOT.2021.3085194.

[23] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. Anwar, "TON_IoT Telemetry Dataset: A New Generation Dataset of IoT and IIoT for Data-Driven Intrusion Detection Systems," *IEEE Access*, vol. 8, pp. 165130–165150, 2020, https://doi.org/10.1109/ACCESS.2020.3022862.

[24] N. Moustafa, M. Keshk, E. Debie, and H. Janicke, "Federated TON_IoT Windows Datasets for Evaluating AI-based Security Applications." arXiv, Oct. 04, 2020, https://doi.org/10.48550/arXiv.2010.08522.

[25] N. Moustafa, M. Ahmed, and S. Ahmed, "Data Analytics-Enabled Intrusion Detection: Evaluations of ToN_IoT Linux Datasets," in *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Guangzhou, China, Sep. 2020, pp. 727–735, https://doi.org/10.1109/TrustCom50675.2020.00100.

[26] N. Moustafa, "New Generations of Internet of Things Datasets for Cybersecurity Applications based Machine Learning: TON_IoT Datasets," in *Proceedings of the eResearch Australasia Conference*, Brisbane, Australia. 2019.

[27] N. Moustafa, "A Systemic IoT-Fog-Cloud Architecture for Big-Data Analytics and Cyber Security Systems: A Review of Fog Computing." arXiv, May 04, 2019, https://doi.org/10.48550/arXiv.1906.01055.

[28] J. Ashraf *et al.*, "IoTBoT-IDS: A novel statistical learning-enabled botnet detection framework for protecting networks of smart cities," *Sustainable Cities and Society*, vol. 72, Sep. 2021, Art. no. 103041, https://doi.org/10.1016/j.scs.2021.103041.

[29] F. Bolikulov, R. Nasimov, A. Rashidov, F. Akhmedov, and Y.-I. Cho, "Effective Methods of Categorical Data Encoding for Artificial Intelligence Algorithms," *Mathematics*, vol. 12, no. 16, Jan. 2024, Art. no. 2553, https://doi.org/10.3390/math12162553.

[30] R. Cassandro, Q. Li, and Z. S. Li, "An Entropy-based Data Reduction Method for Data Preprocessing," in *2023 IEEE International Conference on Prognostics and Health Management (ICPHM)*, Jun. 2023, pp. 351–356, https://doi.org/10.1109/ICPHM57936.2023.10194224.

[31] B. H. Bhavani and N. C. Naveen, "An Approach to Determine and Categorize Mental Health Condition using Machine Learning and Deep Learning Models," *Engineering, Technology & Applied Science Research*, vol. 14, no. 2, pp. 13780–13786, Apr. 2024, https://doi.org/10.48084/etasr.7162.

[32] W. T. Valavan, N. Joseph, and G. U. Srikanth, "Network Intrusion Detection System Based on Information Gain with Deep Bidirectional Long Short-Term Memory," *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 4, pp. 45–56, Aug. 2024, https://doi.org/10.22266/ijies2024.0831.04.

[33] M. Bartouli, A. Msolli, A. Helali, and H. Fredj, "A Real-Time Charge Predictive Model for Intelligent Networks," *Engineering, Technology & Applied Science Research*, vol. 14, no. 5, pp. 17091–17098, Oct. 2024, https://doi.org/10.48084/etasr.7845.

[34] K. A. Nadhum, S. M. Sam, and S. Usman, "Prediction Model Using Deep Learning for Lung Illness Severity Among Covid-19 Patients in Iraq," in *2024 5th International Conference on Smart Sensors and Application (ICSSA)*, Penang, Malaysia, Sep. 2024, pp. 1–6, https://doi.org/10.1109/ICSSA62312.2024.10788660.