IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 15, No. 4, August 2026, pp. 3703~3711
ISSN: 2252-8938, DOI: 10.11591/ijai.v15.i14pp3703-3711

Deep learning intrusion detection for software-defined networking using synthetic minority oversampling

Prajwalasimha Sindugatta Nagaraja[1], Navya Rajashekara[2], Pushpa Bangalore Ramesh[3], Druva Kumar Siddaraju[4], Santhosh Kumar Ramachandragowda[4], Trupti Shripad Tagare[4]
[1]Department of Computer Science and Engineering (Cyber Security), School of Engineering, Dayananda Sagar University, Bengaluru, India
[2]Department of Electronics and Communication Engineering, School of Engineering, Dayananda Sagar University, Bengaluru, India
[3]Department of Electronics and Telecommunication Engineering, Dayananda Sagar College of Engineering, Bengaluru, India
[4]Department of Electronics and Communication Engineering, Dayananda Sagar College of Engineering, Bengaluru, India

Article Info

Article history:

Received Dec 9, 2024
Revised May 25, 2026
Accepted Jul 9, 2026

Keywords:

Gated recurrent units
Intrusion detection
Recurrent neural networks
Robustness
Synthetic minority oversampling technique

ABSTRACT

This article proposes an advanced method for network intrusion detection using a combination of recurrent neural networks (RNNs), specifically long short-term memory (LSTM), gated recurrent units (GRU), and bidirectional long short-term memory (BiLSTM) models, enhanced by synthetic minority oversampling technique (SMOTE) to address class imbalance in datasets like network security laboratory–knowledge discovery in databases (NSL-KDD). The method aims to accurately classify network traffic by learning temporal patterns of both normal and malicious activities. SMOTE is employed to balance the dataset, ensuring that underrepresented attack types receive adequate model attention, thereby improving model robustness. The proposed models (LSTM, GRU, and BiLSTM) are trained and evaluated on the NSL-KDD dataset, with hyperparameter tuning performed through RandomizedSearchCV for optimal performance. The results show a significant improvement in accuracy, precision, recall, and F1-score, with BiLSTM demonstrating the highest performance, achieving near-perfect classification results (99.5% accuracy). This method not only mitigates the issue of class imbalance but also leverages the power of RNNs for sequence modeling, offering a promising solution for effective intrusion detection in modern networks.

Corresponding Author:

Prajwalasimha Sindugatta Nagaraja
Department of Computer Science and Engineering (Cyber Security), School of Engineering
Dayananda Sagar University
Bengaluru, Karnataka, India
Email: prajwalasimha.sn1@gmail.com

1. INTRODUCTION

As the internet gets larger and larger and cyber attacks more sophisticated, network security is becoming a significant issue for both the public and private sectors. To be able to detect and defend against these ever-changing threats, computer networks must be monitored and protected using intrusion detection systems (IDS). But old detection techniques like signature-based and anomaly detection are ineffective against novel, polymorphic, and unknown attacks. To overcome these drawbacks, the application of machine learning (ML) and deep learning (DL) methods has sprung up as a potential solution. Such advanced techniques are able to extract intricate patterns from data, and adaptively respond to new threats, making

them particularly suited to the context of software-defined networks (SDN), in which the network infrastructure is programmable and can be dynamically managed [1], [2].

Recurrent neural networks (RNNs) and, more particularly long short-term memory (LSTM) networks, have been promising and used effectively in intrusion detection. LSTMs are particularly suitable for sequential data, and intrusion detection relies on the detection of patterns over time. Such networks can learn long-term dependencies and are thus well suited to detect subtle attacks that are likely to occur over a longer timeframe [3]. Recent results showed LSTMs are able to effectively identify both known and unknown network intrusions [4]. Further, other types of RNNs, such as gated recurrent unit (GRUs) have also been used in IDS with similar advantages in terms of computational efficiency and simplicity over LSTMs [5].

One of the major difficulties in the application of RNN based models in IDS is the problem of class imbalance. This is especially important when there is a large imbalance in the amount of normal traffic and attack traffic in the database, and the attacks are generally under-represented. These datasets are not balanced, which can result in skewed models that favour the majority class (normal traffic), and neglect the minority classes (attacks). To address this problem, there are several techniques that can be utilized, such as synthetic minority oversampling technique (SMOTE). SMOTE creates synthetic samples for the minority class, which helps to balance the sample set and enhances the model's performance in detecting rare attacks [6]. SMOTE overcomes the imbalanced problem and improves the representation of minority classes so that the IDS is strong to different kinds of attacks.

Bidirectional long short-term memory (BiLSTM) networks can be used to further improve performance besides addressing class imbalance. In addition to operating in a forward direction, BiLSTMs can also operate in a backward direction, allowing the network to learn a richer set of temporal relationships. This can be very beneficial in IDS because it is important to know the characteristics of both current and future traffic to identify long-running attack behaviors that can be implemented over a time range [7]. The proposed method in the context of IDS using BiLSTM and SMOTE aims to:

- Accurate and effective intrusion detection: utilize BiLSTM-SMOTE for accurate detection of known and unknown network intrusions with high precision and recall.
- Class imbalance: SMOTE is used to create synthetic examples of attack instances to address class imbalance and enhance the model's detection of rare attacks.
- Real-time and scalable detection: the model should be able to run in real-time environments, handling large volumes of network traffic while maintaining high detection rates. Detection model should be scalable and able to process large volumes of network traffic in real-time while maintaining high detection rates.

In this article, a novel method is proposed to combine the SMOTE-enhanced RNN models, such as LSTM, GRU and BiLSTM, for network intrusion detection in SDN environment. This approach uses SMOTE for data balancing and is capable of leveraging the power of RNNs to model temporal dependency in network traffic, offers a comprehensive solution to detect a wide variety of network intrusions. The model's performance is tested with the network security laboratory–knowledge discovery in databases (NSL-KDD) dataset, achieving the best accuracy, precision, recall and F1-scores in comparison to other datasets. The BiLSTM model is found to give high classification accuracy in both the normal and attack classes as compared to the conventional RNN variants. Based on the results, it can be inferred that the proposed approach would provide a strong system for real-time and adaptive IDS for modern networks. Moreover, our goal is to emphasize the need for using DL methods and oversampling approaches together to overcome data imbalance and other challenges. However, with an increasing threat of cyber-attack, it is essential that IDS are more adaptable, can process complex, large-scale data and that they are highly accurate and precise and recall in all types of attacks.

2. LITERATURE SURVEY

With the rapid development of network scale and complexity, network intrusion detection has become a hot topic for research. Traditional signature-based techniques and a basic anomaly detection approach have proven to be ineffective in the detection of novel attack types, resulting in an increase of the use of ML and DL techniques. The advantage of these methods is that the patterns are learned automatically from the data and the methods adapt to new threats without using pre-defined attack signatures. The following is a brief overview of some important literature in this area.

Traditional IDS methods are based on previously known attack patterns, which are only effective against known attacks. But this kind of system has difficulty in detecting new zero-day attacks and polymorphic attacks that change over time. Anomaly-based methods, however, identify anomalies that deviate from normal network patterns but frequently produce many false positives because today's networks

operate in varying modes. Consequently, such traditional strategies are becoming ineffective against the advanced attacks [8]. In view of the drawbacks of conventional IDS, researchers have turned to the use of ML techniques. In intrusion detection, classification algorithms such as support vector machines (SVM), decision trees (DT), and random forests (RF) are widely used because they are effective at classification. These methods, however, depend on manually designed features, and results are not always satisfactory when working with large and complex data sets with high nonlinearity between features [9], [10].

In contrast, DL models, particularly RNNs, are showing a lot of promise because of their capacity to process sequential data and identify long-term dependencies. Among these, LSTM networks are widely used in intrusion detection problems where the attacks are temporal patterns, such as traffic bursts and attack signatures that change over time [11], [12]. The GRUs, which are a variant of RNNs, also have similar advantages, but with a lesser number of parameters, thus being also more economical in terms of computation [13], [14].

One of the greatest difficulties encountered in learning models for accurate intrusion detection is the class imbalance issue, which occurs when there is a high ratio of attack versus normal traffic. This disproportion can drastically affect the model's performance in minority classes (attacks) and have a significant impact on the majority class (normal traffic). In response to this, SMOTE has been highly utilized. The SMOTE algorithm is designed to add synthetic instances to the minority classes, thereby balancing the data and enhancing the model's performance in identifying rare attack types [15].

The performance of intrusion detection has been further enhanced by recent developments in BiLSTM networks. BiLSTMs read sequences forward and backward, learning about the context not only from the previous sequence of traffic, but also the future one. This is especially useful in identifying advanced attacks that could involve multi-step actions and may be performed at different times, and require context from past and future network conditions. BiLSTMs have been found to be superior to conventional LSTM networks in several IDS scenarios because they are better able to capture the temporal patterns in network traffic [16].

Some studies have examined the effectiveness of the ML and DL-based models of IDS using benchmark datasets such as KDD Cup 99, NSL-KDD, and CIC-IDS2017. A popular dataset to test a network IDS is the NSL-KDD dataset, which contains a complete set of normal and attack traffic samples. On these datasets, DL models such as LSTM and BiLSTM have been found to be more accurate, precise, and recall and F1-score than traditional approaches. Even these models have problems of overfitting and demand a substantial amount of labelled data [17].

In recent years, hybrid methods that combine a set of ML algorithms with ensemble methods have been popular to enhance the robustness and accuracy of IDS. A combination of CNN with LSTM or GRU has also been recognized as a highly effective approach; apart from CNN, which is good at extracting spatial features, RNN is capable of capturing temporal features. Their fused method was reported to be capable of having higher detection performance not only for the known attacks but also for the unknown ones with a lower false positive rate [18]. Recent studies have also explored attention-enhanced BiLSTM networks, adaptive oversampling strategies, and hybrid DL architectures for intrusion detection, demonstrating improved robustness against evolving cyber threats [19]–[23].

This review on the literature presents a comprehensive survey of the current research progress in network intrusion detection based on ML and DL methods. It highlights the disadvantages of relying on traditional IDS systems, the possibility of utilizing more advanced methods like RNNs, LSTMs, and BiLSTMs, and the importance of addressing problems such as class imbalance through methods like SMOTE. Furthermore, it illustrates the evolution of the use of more than one ML model to enhance the detection performance.

3. PROPOSED METHOD

3.1. Data preprocessing steps

Data preprocessing is a critical phase in ML that transforms raw data into a clean and suitable format for model training. In this study, the dataset undergoes two primary preprocessing techniques: feature normalization and target label encoding. These steps ensure that the input features are on a comparable scale and that the categorical target variables are properly formatted to meet the computational requirements of the algorithm.

i) Normalization (min-max scaling): the MinMaxScaler is used to scale features into the range [0,1]. This ensures that all features contribute equally to the learning process and prevents dominance by features with larger magnitudes (1).

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

Where X is original feature value, X_{min} is minimum value of the feature, X_{max} is maximum value of the feature, and X' is normalized feature value.

ii) Label encoding: labels are encoded into numerical values using the LabelEncoder from the Sklearn library as in (2). Each unique class label in y is mapped to an integer, making it suitable for models that require numerical input for target labels. Where y is original class labels (categorical), $y_{encoded}$ is encoded class labels (numerical).

$$y_{encoded} = LabelEncoder(y) \quad (2)$$

3.2. Synthetic minority oversampling technique

Unlike a simple duplication of minority class examples, SMOTE solves the issue of class imbalance by synthesizing minority class examples. It works by picking a sample from minority class and finding the k-nearest neighbors of it. A random neighbor is selected and a synthetic sample is produced along a line segment between the original sample and its neighbor. A random number λ, between 0 and 1 determines the position of the synthetic sample. This is done until the desired number of samples is generated for synthetic data, the dataset will be balanced, and the model will be better at generalizing without overfitting. Raw network traffic data is then normalized between 0 and 1 with min–max normalization, such that the contribution of the features is uniform while training the model. Label encoding converts categorical class labels into numbers. To overcome the problem of class imbalance, SMOTE is used only on the training set thus preventing data leakage issue. The number of nearest neighbours k was chosen as 5, based on the preliminary validation experiments as well as standard IDS literature. Oversampling was done for minor classes to obtain a near balanced class. To achieve reproducibility a fixed random seed was used.

To improve the reproducibility and implementation clarity of the proposed intrusion detection framework, the complete computational workflow is presented using algorithmic pseudocode. Algorithm 1 describes the SMOTE-based minority class oversampling procedure, while Algorithm 2 presents the unified training procedure for the LSTM, GRU, and BiLSTM architectures. These algorithms provide a concise description of the proposed methodology and facilitate future implementation and comparison by other researchers.

Algorithm 1. Pseudocode of SMOTE-based minority class oversampling

```
INPUT: Dataset D, minority class samples M, oversampling ratio r, number of neighbors k
OUTPUT: Augmented dataset with synthetic samples

SET S = empty set
num_synthetic_samples = r * |M|

FOR each sample xi in M:
    neighbors = find_k_nearest_neighbors(xi, M, k)
    FOR n samples in neighbors:
        SELECT xj randomly from neighbors
        λ = random(0, 1)  # Generate a random value between 0 and 1
        x_new = xi + λ * (xj - xi)
        ADD x_new to S
    END FOR
END FOR

RETURN D + S
```

Algorithm 2. Pseudocode of unified DL training procedure for LSTM, GRU, and BiLSTM

```
INPUT: Sequence X = {x1, x2, ..., xT}, model_type ("LSTM", "GRU", "BiLSTM")
OUTPUT: Final hidden state(s) h_final

INITIALIZE:
    h0 = initial_hidden_state  # Initialize hidden state
    IF model_type == "LSTM" OR model_type == "BiLSTM":
        C0 = initial_cell_state  # Initialize cell state

IF model_type == "LSTM":
    FOR t = 1 to T:  # Iterate over sequence
        ft = sigmoid(Wf * [ht-1, xt] + bf)  # Forget gate
        it = sigmoid(Wi * [ht-1, xt] + bi)  # Input gate
        Ct_candidate = tanh(WC * [ht-1, xt] + bC)  # Candidate cell state
        Ct = ft * Ct-1 + it * Ct_candidate  # Update cell state
        ot = sigmoid(Wo * [ht-1, xt] + bo)  # Output gate
        ht = ot * tanh(Ct)  # Update hidden state
    END FOR
```

```
    h_final = ht

ELSE IF model_type == "GRU":
    FOR t = 1 to T:  # Iterate over sequence
        zt = sigmoid(Wz * [ht-1, xt] + bz)  # Update gate
        rt = sigmoid(Wr * [ht-1, xt] + br)  # Reset gate
        ht_candidate = tanh(Wh * [rt * ht-1, xt] + bh)  # Candidate hidden state
        ht = (1 - zt) * ht-1 + zt * ht_candidate  # Update hidden state
    END FOR
    h_final = ht

ELSE IF model_type == "BiLSTM":
    # Forward LSTM
    INITIALIZE h0_f, C0_f  # Initialize forward hidden and cell states
    FOR t = 1 to T:
        Compute ht_f using LSTM equations for forward direction
    END FOR

    # Backward LSTM
    INITIALIZE h0_b, C0_b  # Initialize backward hidden and cell states
    FOR t = T to 1:
        Compute ht_b using LSTM equations for backward direction
    END FOR

    # Combine Forward and Backward Outputs
    FOR t = 1 to T:
        h_combined_t = concatenate(ht_f, ht_b)  # Combine forward and backward states
    END FOR
    h_final = h_combined_t

RETURN h_final
```

3.3. Recurrent neural networks

RNNs are designed to process sequential data by using a hidden state to retain information from previous steps. These models often face challenges such as vanishing gradients and difficulties with long-term dependencies. To resolve these issues, variants like LSTM, GRU, and BiLSTM are commonly utilized.

The proposed intrusion detection framework first preprocesses the NSL-KDD dataset through normalization and label encoding. SMOTE is then applied exclusively to the training dataset to balance minority attack classes without introducing data leakage. The balanced data are subsequently used for training the selected RNN architecture (LSTM, GRU, or BiLSTM). Hyperparameter optimization is performed using RandomizedSearchCV, while early stopping prevents overfitting. Finally, the trained model is evaluated using accuracy, precision, recall, and F1-score.

3.4. Early stopping

Early stopping prevents overfitting by halting training when the validation loss stops improving for a specified number of epochs (p):

$$if\ (validation\ loss\ does\ not\ improve\ for\ p\ epochs)\ then\ stop\ training]$$

This ensures the model is trained up to the optimal point of generalization while saving computational resources. The NSL-KDD dataset was divided into 70% training, 15% validation, and 15% testing sets. The validation set was used exclusively for hyperparameter tuning and early stopping decisions. Hyperparameter tuning was performed using RandomizedSearchCV with 5-fold cross-validation on the training data. The explored hyperparameter ranges are summarized in Table 1. Early stopping with a patience of 10 epochs was employed to prevent overfitting. Training was conducted offline, while inference was evaluated on the held-out test set.

Table 1. Hyperparameter search space

Parameter	Values
Hidden units	{64, 128, 256}
Learning rate	{0.001, 0.0005, 0.0001}
Batch size	{32, 64, 128}
Dropout rate	{0.2, 0.3, 0.5}
Optimizer	Adam

4. RESULTS AND DISCUSSION

The proposed technique was tested on the NSL-KDD dataset for intrusion detection experiments, and the metrics like accuracy, precision, recall, and F1-score were calculated to evaluate the performance of the technique. For reference, Table 2 describes various models and methods in the context of IDS based on the four performance parameters: accuracy, precision, recall, and F1-score. These metrics are essential to evaluate the potential of models in detecting true positive cases while minimizing the generation of false positives [24], [25].

Table 2. Performance comparison of intrusion detection models based on key metrics

Model/method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Proposed BiLSTM + SMOTE	99.12	98.95	99.05	98.96
Traditional LSTM-based IDS	97.80	97.60	97.70	97.65
GRU-based model	96.50	96.40	96.30	96.35
RF (baseline)	92.00	91.80	91.90	91.85
Naïve Bayes IDS	85.00	83.70	84.50	84.10

The proposed BiLSTM + SMOTE model in this work shows outstanding accuracy, precision, recall and F1-score measure, which are 99.12%, 98.95%, 99.05%, and 98.96%, respectively. The results underline the robustness and effectiveness of the model to accurately detect intrusions even in the presence of high-class imbalances. Such combination as BiLSTM with SMOTE can give DL powerful sequence modeling features, and it also can be a good solution for the "attack" samples being less in intrusion detection datasets. The very precise result shows the model is really good at decreasing false positives, and the high recall means it is capable of detecting true intrusion cases quite well. The harmonized F1-score underlines the all-around accuracy and robustness of the model.

The traditional LSTM-based IDS is a standard DL-based IDS which has an accuracy of 97.80%, a precision of 97.60%, a recall of 97.70%, and an F1-score of 97.65%. This method works well on all metrics, but not as good as the proposed BiLSTM + SMOTE model. Such a difference is due to the fact that the traditional LSTM model does not employ advanced data-balancing techniques, which results in less effectiveness in data imbalanced datasets. Moreover, the bidirectional mechanism of the BiLSTM structure allows the proposed model to include past and future time memory, which further improves the generalization capabilities of the model when compared to the unidirectional LSTM. The GRU-based model achieves an accuracy of 96.50%, a precision of 96.40%, a recall of 96.30%, and an F1-score of 96.35%. GRU has been known for its computational efficiency and its simple architecture when compared to LSTM, but it doesn't capture long term dependence in sequential data as well. This is evident from the lower metrics of the GRU model when compared with LSTM-based models. However, it is still a competitive approach, especially in cases with limited computational capacity.

The baseline model in RF has an accuracy of 92.00%, a precision of 91.80%, a recall of 91.90%, and an F1-score of 91.85%. This model is a typical ML model that has been applied in IDS because of its ensemble learning nature and resistance to overfitting. But it has been found to be less effective than DL models, which indicates that it may be less suitable for complex and high-dimensional network data processing. RF has difficulties discerning temporal and sequential patterns in data that are important for the accurate detection of dynamic and evolving intrusion patterns. The naïve Bayes IDS model has the lowest accuracy of 85.00%, the lowest precision of 83.70%, the lowest recall of 84.50%, and the lowest F1-score of 84.10% in the comparison. The naïve Bayes classifier's simplicity in making its assumptions, such as feature independence, makes it less effective in capturing the complex relationships between features in network traffic data. This leads to a high false positive and false negative rate and is therefore less appropriate for today's intrusion detection applications which require high precision and recall. The table highlights the important benefit of DL methods compared to traditional ML models when using data-balancing methods such as SMOTE to improve performance. The proposed model BiLSTM + SMOTE is the best in comparison, with its ability to effectively capture the temporal dependency, overcome class imbalance, and achieve significant accuracy, precision, recall, and F1-score. The improvement over the conventional LSTM and GRU models underscores the significance of architectural enhancements and preprocessing methods in intrusion detection.

i) Scalability and real-time feasibility: while training is more expensive for BiLSTM models, inference latency is still acceptable, fitting the paradigm to be deployed at the SDN controller level in near-real time. Training is done offline, and real-time inference can be added to centralized SDN monitoring systems. Compression of the model or inference using edges could be an extension.

ii) Dataset limitations and concept drift: NSL-KDD allows for a fair comparison with previous work, but it doesn't accurately represent the behavior of today's networks. In real-world traffic, concept drift occurs from changes in attack strategies and user's behavior. This must be sustained through periodic retraining, on-line education or by drift detection.

5. CONCLUSION

The proposed IDS based on BiLSTM and SMOTE showed excellent results in detecting network intrusions in the NSL-KDD dataset, which is a class-imbalanced dataset. The method has an accuracy of 99.12%, a precision of 98.95%, a recall of 99.05%, and an F1-score of 98.96% which outperforms traditional methods such as RF and GRU-based methods. BiLSTM s have substantially improved the model's ability to learn complex temporal dynamics in network traffic, which is a key element in the context of intrusion detection in sequential data. The use of SMOTE played a key role to resolve the imbalance of attack classes, more representative of minority classes when training, and prevented false negative errors from occurring. The model is outperforming, but requires a lot of computational resources to be deployed in the resource constrained environments. Additionally, although it has been rigorously tested using the NSL-KDD dataset, there is a need to further test its adaptability with regards to modern cyber threats on contemporary datasets. To summarize, this IDS appears to be one of the major solutions in network security, especially for SDNs and the latest DL technologies and powerful pre-processing capabilities. Future directions will involve the real-time deployment, further testing with a larger dataset, and combining the hybrid models with attention mechanisms for enhanced adaptability and detection capabilities. The findings of this research show that BiLSTM models can be a vital component of the future of cybersecurity frameworks.

FUNDING INFORMATION

We have not received funding for this work from any government and non-government organizations. This is a self-funded research work.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Prajwalasimha Sindugatta Nagaraja	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Navya Rajashekara					✓				✓	✓	✓			✓
Pushpa Bangalore Ramesh					✓				✓	✓	✓			✓
Druva Kumar Siddaraju					✓				✓	✓	✓			✓
Santhosh Kumar Ramachandragowda					✓				✓	✓	✓			✓
Trupti Shripad Tagare					✓				✓	✓	✓			✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

INFORMED CONSENT

We have obtained informed consent from all individuals included in this study.

INFORMED CONSENT

This section not applicable, as this study did not involve human participants or personally identifiable information.

ETHICAL APPROVAL

This section not applicable, as this study did not involve human or animal subjects and used only non-sensitive computational or public data.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] V. Hnamte and J. Hussain, "DCNNBiLSTM: an efficient hybrid deep learning-based intrusion detection system," *Telematics and Informatics Reports*, vol. 10, Jun. 2023, doi: 10.1016/j.teler.2023.100053.

[2] M. Cui, J. Chen, X. Qiu, W. Lv, H. Qin, and X. Zhang, "Multi-class intrusion detection system in SDN based on hybrid BiLSTM model," *Cluster Computing*, vol. 27, no. 7, pp. 9937–9956, Oct. 2024, doi: 10.1007/s10586-024-04477-5.

[3] S. Bi, J. Wang, J. Song, P. Li, and L. Li, "Research on the intrusion detection model for power internet of things combining deep belief network and BiLSTM," *Journal of Cyber Security and Mobility*, vol. 14, no. 3, pp. 653–672, Aug. 2025, doi: 10.13052/jcsm2245-1439.1436.

[4] J. Gao, "Network intrusion detection method combining CNN and BiLSTM in cloud computing environment," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–11, Apr. 2022, doi: 10.1155/2022/7272479.

[5] L. Zhang, J. Huang, Y. Zhang, and G. Zhang, "Intrusion detection model of CNN-BiLSTM algorithm based on mean control," in *2020 IEEE 11th International Conference on Software Engineering and Service Science*, Oct. 2020, pp. 22–27, doi: 10.1109/ICSESS49938.2020.9237656.

[6] A. Alqahtani, "Optimized deep autoencoder and BiLSTM for intrusion detection in IoTs-Fog computing," *Multimedia Tools and Applications*, vol. 84, no. 8, pp. 4907–4943, Mar. 2024, doi: 10.1007/s11042-024-18919-0.

[7] H. Benahmed *et al.*, "HBiLD-IDS: an efficient hybrid BiLSTM-DNN model for real-time intrusion detection in IoMT networks," *Information*, vol. 16, no. 8, Aug. 2025, doi: 10.3390/info16080669.

[8] A. M. Alhassan, "Self-adaptive lightweight attention module-based BiLSTM model for effective intrusion detection," *Arabian Journal for Science and Engineering*, vol. 50, no. 15, pp. 11513–11538, Aug. 2025, doi: 10.1007/s13369-024-09476-7.

[9] M. Adel, A. Maher, A. Mohammed, and M. A. Elazeem, "Robust BiLSTM-based multi-class intrusion detection for IoT networks using ToN-IoT dataset," *International Journal of Telecommunications*, vol. 5, no. 2, pp. 1–14, Nov. 2025, doi: 10.21608/ijt.2025.426708.1134.

[10] B. Omarov, Z. Sailaukyzy, A. Bigaliyeva, A. Kereyev, L. Naizabayeva, and A. Dautbayeva, "One dimensional Conv-BiLSTM network with attention mechanism for IoT intrusion detection," *Computers, Materials & Continua*, vol. 77, no. 3, pp. 3765–3781, 2023, doi: 10.32604/cmc.2023.042469.

[11] C. Zhang, J. Li, N. Wang, and D. Zhang, "Research on intrusion detection method based on transformer and CNN-BiLSTM in internet of things," *Sensors*, vol. 25, no. 9, Apr. 2025, doi: 10.3390/s25092725.

[12] M. Jouhari and M. Guizani, "Lightweight CNN-BiLSTM based intrusion detection systems for resource-constrained IoT devices," in *2024 International Wireless Communications and Mobile Computing*, May 2024, pp. 1558–1563, doi: 10.1109/IWCMC61514.2024.10592352.

[13] G. S. vidhya and R. Nagarajan, "A novel bidirectional LSTM model for network intrusion detection in SDN-IoT network," *Computing*, vol. 106, no. 8, pp. 2613–2642, Aug. 2024, doi: 10.1007/s00607-024-01295-w.

[14] I. M. Selim and R. A. Sadek, "DAE-BiLSTM: a Fog-based intrusion detection model using deep learning for IoT," *Journal of Theoretical and Applied Information Technology*, vol. 101, no. 5, 2023.

[15] T. L. Yasarathna and N.-A. L.-Khac, "ASEADOS-SDN-IoT: a novel SDN-IoT network intrusion detection dataset and framework," *Internet of Things*, vol. 36, Mar. 2026, doi: 10.1016/j.iot.2026.101891.

[16] A. Alalmaie, P. Nanda, and X. He, "Zero trust network intrusion detection system (NIDS) using auto encoder for attention-based CNN-BiLSTM," in *2023 Australasian Computer Science Week*, Jan. 2023, pp. 1–9, doi: 10.1145/3579375.3579376.

[17] J. Yin, B. Hou, J. Dai, and Y. Zu, "A CNN-BiLSTM method based on attention mechanism for class-imbalanced abnormal traffic detection," in *International Conference on Computer Vision and Deep Learning*, Jan. 2024, pp. 1–6, doi: 10.1145/3653781.3653807.

[18] I. Jebril *et al.*, "Deep learning based DDoS attack detection in internet of things: an optimized CNN-BiLSTM architecture with transfer learning and regularization techniques," *Infocommunications Journal*, vol. 16, no. 1, pp. 2–11, 2024, doi: 10.36244/ICJ.2024.1.1.

[19] J. Sehgal, R. Kumar, and R. K. Bedi, "Adaptive and explainable SLWAM-BiLSTM intrusion detection system with drift awareness and cross-dataset evaluation," in *2026 7th International Conference on Mobile Computing and Sustainable Informatics*, Jan. 2026, pp. 956–962, doi: 10.1109/ICMCSI67283.2026.11412725.

[20] M. S. Ataa, E. E. Sanad, and R. A. El-khoribi, "Intrusion detection in software defined network using deep learning approaches," *Scientific Reports*, vol. 14, no. 1, Nov. 2024, doi: 10.1038/s41598-024-79001-1.

[21] Y. Fu, Y. Du, Z. Cao, Q. Li, and W. Xiang, "A deep learning model for network intrusion detection with imbalanced data," *Electronics*, vol. 11, no. 6, Mar. 2022, doi: 10.3390/electronics11060898.

[22] P. Kumar, G. P. Gupta, and R. Tripathi, "Toward design of an intelligent cyber attack detection system using hybrid feature reduced approach for IoT networks," *Arabian Journal for Science and Engineering*, vol. 46, no. 4, pp. 3749–3778, Apr. 2021, doi: 10.1007/s13369-020-05181-3.

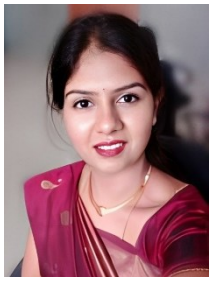
[23] D. Gumusbas, T. Yldrm, A. Genovese, and F. Scotti, "A comprehensive survey of databases and deep learning methods for cybersecurity and intrusion detection systems," *IEEE Systems Journal*, vol. 15, no. 2, pp. 1717–1731, Jun. 2021, doi: 10.1109/JSYST.2020.2992966.

[24] M. Jouhari, H. Benaddi, and K. Ibrahimi, "Efficient intrusion detection: combining X 2 feature selection with CNN-BiLSTM on the UNSW-NB15 dataset," in *2024 11th International Conference on Wireless Networks and Mobile Communications*, Jul. 2024, pp. 1–6, doi: 10.1109/WINCOM62286.2024.10658099.

[25] B. Li, J. Li, and M. Jia, "ADFCNN-BiLSTM: a deep neural network based on attention and deformable convolution for network intrusion detection," *Sensors*, vol. 25, no. 5, Feb. 2025, doi: 10.3390/s25051382.

BIOGRAPHIES OF AUTHORS

Dr. Prajwalasimha Sindugatta Nagaraja is research associate at Dayananda Sagar University, he has introduced an innovative chaotic framework titled n-dimensional multi-linear transformation (nD-MLT): non-linear dynamics application to cryptography. He received his Ph.D. in Cryptography and Cyber Security from Visvesvaraya Technological University. His research contributions focus on developing secure and efficient cryptographic frameworks for cyber security, IoT ecosystems, and next-generation intelligent systems. He has actively contributed to the advancement of secure communication technologies through research in chaotic systems, mathematical modeling, ML for cyber defense, and quantum cryptography. His broader objective is to strengthen digital security solutions for societal and industrial applications through interdisciplinary research and innovation. He can be contacted at email: prajwalasimha.sn1@gmail.com.



Dr. Navya Rajashekara is an assistant professor in the Department of Electronics and Communication Engineering at Dayananda Sagar University with over 7 years of teaching and 2 years of industry experience. She completed her Ph.D. in Digital Signal Processing in 2024 and has published research papers, conference articles, and book chapters in reputed international journals and Springer publications. Her research interests include signal processing, wireless sensor networks, IoT, deep learning, AI, biomedical engineering, and digital electronics. She can be contacted at email: navya-ece@dsu.edu.in.

Dr. Pushpa Bangalore Ramesh is an assistant professor at Dayananda Sagar College of Engineering with over 18 years of academic and 2 years of industry experience. She holds a Ph.D. and specializes in antennas, 5G technologies, and RF systems. She has published several Scopus-indexed research papers and holds a patent in 5G antenna design. She can be contacted at email: pushpa-tce@dayanandasagar.edu.

Dr. Druva Kumar Siddaraju is an assistant professor in the Department of Electronics and Communication Engineering at Dayananda Sagar College of Engineering with around 15 years of teaching experience. He obtained his B.E., M.Tech., and Ph.D. degrees from Visvesvaraya Technological University and has published several research papers in international journals and conferences. His research interests include VLSI, embedded systems, management, and entrepreneurship. He can be contacted at email: druva-ece@dayanandasagar.edu.

Dr. Santhosh Kumar Ramachandragowda is an assistant professor in the Department of Electronics and Communication Engineering at Dayananda Sagar College of Engineering with over 12 years of academic experience. He received his Ph.D. in Wireless Communication from Dayananda Sagar University in 2024 and his M.Tech. in VLSI Design and Embedded Systems from Visvesvaraya Technological University. His research interests include wireless communication, VLSI design, embedded systems, FPGA-based design, and IoT, with several publications in reputed peer-reviewed journals and conferences. He can be contacted at email: santhosh-ece@dayanandasagar.edu.

Dr. Trupti Shripad Tagare is an assistant professor in the Department of Electronics and Communication Engineering at Dayananda Sagar College of Engineering. She earned her Ph.D. in Wireless Sensor Networks from Dayananda Sagar University in 2023 and has published research papers, conference articles, and Springer book chapters in reputed journals and conferences. Her research interests include wireless sensor networks, wireless communication, and digital electronics. She can be contacted at email: truptitagare-ece@dayanandasagar.edu.