An Enhanced Intrusion Detection System Using Attention-Based Stacked Sparse Autoencoder Feature Extraction

Venkata Ramani Varanasi
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India
varanasivenkataramani@gmail.com (corresponding author)

Shaik Razia
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India
razia28sk@gmail.com

Received: 19 March 2025 | Revised: 7 April 2025 | Accepted: 12 April 2025

 DOI: https://doi.org/10.48084/etasr.11034

ABSTRACT

Attention-based stacked sparse autoencoders (AB-SSAEs) are an innovative method for improving Intrusion Detection Systems (IDSs) through the extraction of important features in high-dimensional and heterogeneous data. The proposed AB-SSAE presents an innovative approach to optimizing feature extraction processes using attention mechanisms and a hierarchy of focused sparse autoencoders. The AB-SSAE architecture employs several layers of sparse autoencoders, which transform features through attention mechanisms at every level, improving precision for feature extraction. AB-SSAE employs adaptive denoising with median filtering as a preprocessing step. From the mined data, normal and intrusion attempts are efficiently classified using a Bidirectional Long-Short-Term Memory (Bi-LSTM) network. The proposed technique was compared with several existing approaches, and the results showed that it can differentiate between malicious and benign network traffic with an accuracy of over 0.98.

Keywords-intrusion detection systems; attention-based stacked sparse autoencoder; median filtering; LSTM

I. INTRODUCTION

The number of individuals and organizations that depend on networks has increased significantly due to technological growth. As a large number of platforms and applications are connected to networks, data tends to be more vulnerable to malicious attacks [1]. The rapid emergence of Internet services and the huge amount of data traffic cause increased security concerns [2]. Network security has become a significant research area and diverse ideas and techniques have been presented to deal with network security concerns. Computer networks can be widely protected using an approach called Intrusion Detection Systems (IDSs). IDSs are considered one of the most prominent approaches for quickly detecting and addressing network intrusions [3-4]. IDSs employ two detection approaches based on anomalies and signatures [5]. Pattern recognition is used to detect attacks in signature-based IDSs, where malware and attacks can be recognized using predefined signatures [6-7]. Through deep analysis of transmitted data, unknown attacks and malware can be identified through anomaly-based IDSs by examining network traffic [8]. In addition, IDSs can detect possible attacks and unauthorized access to the network.

Diverse forms of attacks are emerging in the networking domain, such as Denial of Service (DoS), Distributed DOS (DDoS), User-to-Root (UtR), Cross-Site Scripting (CSS), Root-to-Local (RtL), and Man-in-the-Middle (MiM) attacks [9-10]. As intrusions increase rapidly every day, they lead to a large number of privacy violations, financial losses, and illegal access to data [11]. Intrusions can be in the form of insider or outsider attacks [12-13]. Security researchers have conducted various research efforts in the anomaly detection domain on the Internet and computer networks. From an application perspective, existing methods suffer from certain drawbacks. Most researchers have suggested approaches based on Machine Learning (ML) [14] and Deep Learning (DL) [15].

ML-based approaches, such as Naive Bayes (NB), Support Vector Machines (SVM), Random Forest (RF), and K-Nearest Neighbor (KNN) classifiers, have been used to identify and detect network intrusions. However, these approaches have a high computational cost, lack a deeper examination of the dataset, and tend to increase false predictions [16]. Contrary to traditional ML-based approaches, DL has gained more attention in recent years, as it is superior to ML [17]. DL methods such as Artificial Neural Networks (ANN), Recurrent

Neural Networks (RNN), and Convolutional Neural Networks (CNN) can promote better data representation and attack detection accuracy. Efficient DL methods can achieve better intrusion detection, minimize error rates, and provide greater training ability.

In [18], a two-phase hybrid IDS was based on a Deep Neural Network (DNN) and a Sparse Autoencoder (SAE). In the initial phase, unsupervised SAE with 11 smoothed regularizations was applied to impose AE sparsity. These 11 smoothed regularizations could help to learn the sparse representation of features. In the second phase, the DNN was applied to detect and classify attacks based on the extracted features. This two-phase model achieved better performance in terms of low false positives and better detection rates. In [19], a DL-based intrusion detection technique was proposed employing a Pre-Trained model with a Deep Autoencoder (PTDAE) associated with a DNN for attack classification. This technique used an automated hyperparameter optimization process that integrated random search and grid search. Three feature extraction models, namely DAE, AE, and SAE, were applied during the pre-training phase, and the best results were achieved with DAE.

In [20], a sophisticated and scalable DL-based IDS was proposed to improve the classification of multiclass attack patterns. This model used LSTM and a Fully Connected Network (FCN) to classify malicious and benign cases and achieved better classification results in five- and two-class cases. Performance was analyzed using various intrusion detection datasets, namely NITRIDS, Kyoto, KDDCorrected, GureKDD, NSLKDD, and KDDCup99, with accuracies of 99.64%, 100%, 99.36%, 99.03%, 98.94%, and 98.52%, respectively. In [21], a Random Neural Network-based Adversarial (RaNN-ADV) IDS was proposed. To generate adversarial attacks, RaNN-ADV used the Jacobian Saliency Map Attack (JSMA) algorithm, which determined the features that offered more variation to benign instances with less added perturbation. RaNN-ADV was trained and tested with benchmark adversarial-only data and NSL-KDD data. When using adversarial-only data, it achieved low classification accuracy due to class imbalance and could be enhanced using a proper feature extraction scheme. However, JSMA is practicable to craft adversarial samples because it varies only some features and includes perturbations that create suitable options for use in limited resource cases.

In [22], a new intrusion detection scheme was proposed, which integrated an Improved Conditional Variational AE (ImCVAE) with a DNN (ImCVAE-DNN). ImCVAE was employed to learn and discover effective sparse representations between classes and data features of the network. Regarding particular intrusion classes, trained ImCVAE could construct new attack instances to maximize training diversity and balance the training data, improving the detection rate of imbalanced attacks. Moreover, the trained ImCVAE was used not only to minimize data dimensionality but also to adjust DNN weights. Thus, DNN could simply achieve global optimization by fine-tuning and backpropagation. In [23], an efficient model was proposed that extracted significant features to address the class imbalance problem. A DL-based classifier was then utilized for intrusion classification. An AE architecture was proposed to learn the specified features. The CICIDS 2017 dataset was utilized to evaluate performance, where the model achieved a recall of 97.41% in detecting BoT attacks and 97.25% in port scan attacks. The performance of web attacks using diverse feature vectors was also analyzed.

In [24], an Asymmetric Parallel AE (APAE) was presented to determine diverse attacks in IoT networks. This architecture consisted of two parallel encoders, each having three successive layers of convolutional filters. This model was lightweight to detect real-time attacks, promoting better generalization performance. The effectiveness of the proposed APAE model was evaluated on the UNSW-NB15, KDDCup99, and CICIDS2017 datasets, and the results showed better accuracy, but more preference was given to the majority class. In [25], a DL-based approach was proposed for attack detection based on an LSTM model. Mutual Information (MI) and Principal Component Analysis (PCA) were employed to reduce dimensionality. The performance of the models was evaluated using the KDD99 dataset in both binary and multiclass classification, with the results showing that the PCA-based models achieved better accuracy in both cases.

A. *Objectives*

The objectives of this study were:

- Preprocess data through M-square normalization, delete duplicates and data without variance or with irrelevant tags to obtain data with better quality.
- Extract the relevant data features using an attention-assisted sparse autoencoder to minimize computational time and maximize accuracy.
- Evaluate intrusion detection performance using various metrics and compare with different baseline methods.

II. PROPOSED METHOD

This study presents a novel intrusion detection mechanism for data communication networks. Initially, data were collected and preprocessed through M-square normalization, and duplicates or data without variance and irrelevant tags were deleted. From the preprocessed data, the data dimensionality was reduced through a feature extraction process using the AB-SSAE to minimize execution time and enhance classification accuracy. From the extracted features, normal and intrusion occurrences were classified through a Bidirectional Long-Short-Term Memory (Bi-LSTM) network.

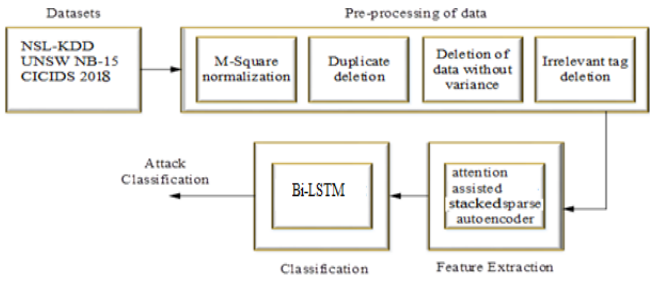

Fig. 1. System architecture.

A. Data Collection

Three datasets were selected: UNSWNB15 [26], NSL KDD [27], and CSE-CIC-IDS 2018 [28]. The CSE-CIC-IDS2018 dataset stands out as a valuable resource for network traffic analysis, offering a diverse collection of benign and common attack scenarios and providing a realistic representation of real-world network traffic. UNSWNB15 has various modern attack types and is well-suited for real-time attack detection. The NSL KDD dataset is useful for benchmark comparisons. These rich datasets allow researchers and practitioners to analyze network behaviors, detect anomalies, and develop robust intrusion detection and prevention systems.

B. Data Preprocessing

In preprocessing, duplicate rows and rows containing null or infinity values were removed. Data were normalized, irrelevant tags were deleted, and SMOTE was used to balance data for binary or multiclass classification. The train test split used a ratio of 80:20.

C. AB-SSAE Architecture

The AB-SSAE architecture comprises multiple layers of sparse autoencoders with attention mechanisms integrated at each layer to improve feature extraction. The model comprises an input layer followed by several hidden layers, each comprising an SAE component. An attention mechanism is incorporated within each layer to dynamically adjust the importance of input features based on their relevance. This attention mechanism helps the model focus on informative features while suppressing irrelevant ones, improving the overall feature representation. This architecture enables hierarchical feature learning, allowing the model to capture complex data patterns. Figure 2 illustrates the architecture of the stacked autoencoder model, which comprises multiple layers of encoding and decoding units.

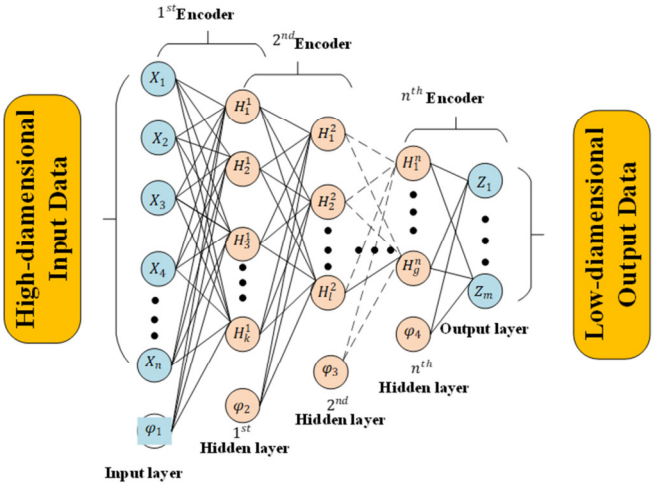

Fig. 2. The structure of the stacked autoencoder model.

Each layer of the encoder network learns increasingly abstract representations of the input data, while the corresponding layer of the decoder network reconstructs the input from these representations. The stacked architecture allows for the extraction of hierarchical features, capturing both low- and high-level data patterns. The greedy layer-wise pretraining method is used to progressively set up each SSAE layer to acquire permission to the improved affiliation loads and bias values of the entire network. Then, backpropagation is used to fine-tune the SSAE until the result of the error function between the input and the output data satisfies the expected requirements to acquire the optimal parameter model.

III. EXPERIMENTS, RESULTS, AND DISCUSSION

The experimental results of AB-SSAE for feature extraction in IDS reveal promising results. Through extensive experimentation, the AB-SSAE demonstrated its effectiveness in capturing meaningful features from network traffic data, enabling improved detection of intrusions. The results indicate that the AB-SSAE architecture successfully learns hierarchical representations of network behavior, allowing it to identify subtle patterns indicative of malicious activity while limiting misleading cases. Table I displays the configuration of the simulation system.

TABLE I. SIMULATION SYSTEM CONFIGURATION

Python Jupiter	Version 3.8.0
Operation system	Ubuntu
Memory capacity	4GB DDR3
Processor	Intel Core i5 @ 3.5GHz

A. Performance of the Proposed Model on CICIDS-2018

Figure 3 shows the training and validation losses of AB-SSAE over epochs on the CICIDS-2018 dataset. In general, the loss decreases as the model is learning. In the early epochs, the training loss tends to decrease rapidly as the model learns basic patterns in the data. As training continues, the training loss becomes flatter, indicating that the model is learning finer details.

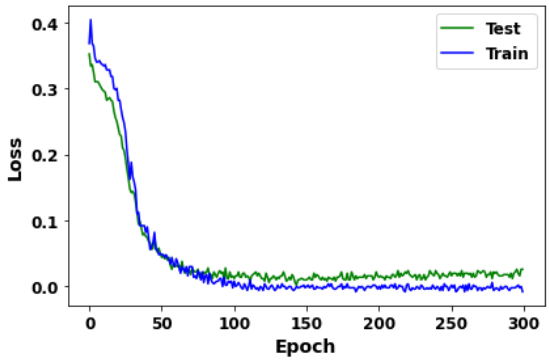

Fig. 3. Training and validation losses of AB-SSAE on CSE-CIC-IDS 2018.

Figure 4 plots the training and validation accuracy over epochs for the proposed IDS using AB-SSAE. Training accuracy is consistently high after the initial epochs, approaching 1.0, indicating that the model fits the training data well. Validation accuracy also increases and reaches stability over epochs, indicating improved generalization and ensuring reliable detection of network intrusions.

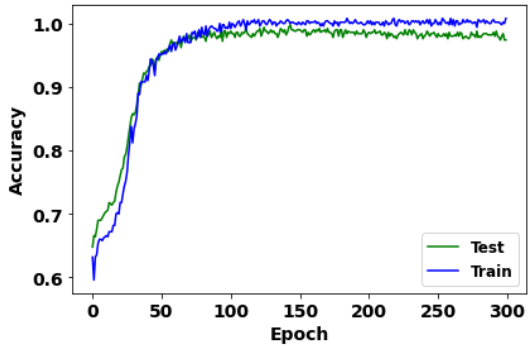

Fig. 4. Training and validation accuracy over epochs on CSE-CIC-IDS 2018.

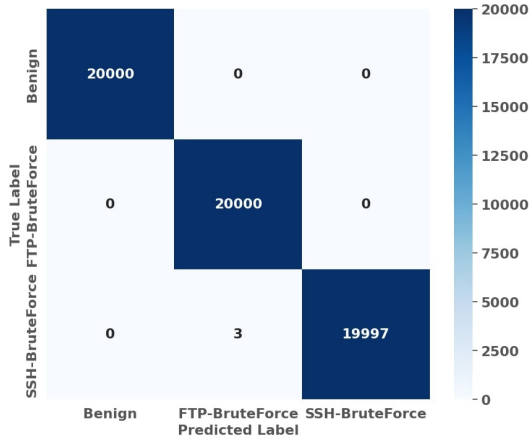

Fig. 5. Confusion matrix for classification performance on CSE-CIC-IDS 2018.

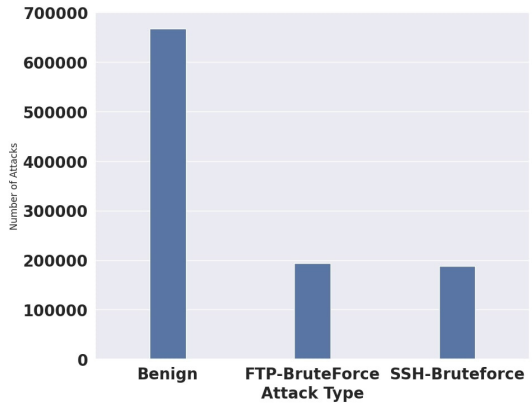

Fig. 6. Number of detected attacks by type.

Figure 5 shows a confusion matrix on a set of data. Each row denotes the original class of a case, and each column denotes the class predicted by the model. The cells on the diagonal show the number of occurrences that were correctly classified, while the off-diagonal cells show the number of cases that were misclassified. The model appears to have performed well on benign traffic, correctly classifying 20000 instances. The model also performed well on FTP-Brute-Force attacks, correctly classifying 20000 instances. There were only three misclassified SSH-Brute-Force instances, which were classified as FTP-Brute-Force attacks, but the model appears to have performed well on SSH-Bruteforce traffic, correctly classifying 19997 instances. Figure 6 shows the number of attacks detected by the proposed IDS for different types. The most common appears to be benign traffic, with more than 700,000 detected samples, by FTP-Bruteforce and SSH-Bruteforce attacks with approximately 200,000 samples detected. There are significantly fewer attacks of the other types detected, including Scan, Probe, and Infiltration.

B. Performance Comparison

By leveraging attention mechanisms and hierarchical feature learning, the proposed model outperformed conventional approaches in detecting and classifying network intrusions. Compared to baseline methods, such as rule-based systems or shallow learning algorithms, the attention-based approach demonstrated enhanced accuracy, robustness, and adaptability to evolving cyber threats. Figure 7 shows an accuracy comparison of the proposed method with Support Vector Machine (SVM) and K-Nearest Neighbors (KNN), indicating significant differences in performance. SVM achieved an accuracy of 0.87, indicating its proficiency in correctly classifying network traffic. KNN followed with an accuracy of 0.78, suggesting slightly lower effectiveness in classification tasks. In contrast, the proposed method exhibited an impressive accuracy of 0.98, showcasing its remarkable ability to accurately discern between benign and malicious network activity. Figure 8 shows an accuracy comparison among LSTM, CNN, RNN, DNN, AE, and the proposed method on the three datasets. The results indicate an almost similar performance of 0.98, which exhibits the impressive performance of the AB-SSAE for feature extraction in IDSs.

Figure 9 shows a comparison of precision among different DL methods on the three datasets. Precision refers to the proportion of true positives out of all positive predictions (avoiding false alarms). Figure 10 shows a comparison of recall among different deep learning methods on the three datasets. Recall refers to the proportion of true positives out of all actual positive cases (avoiding missed detections). Figure 11 shows an F1-score comparison of the same models. The proposed model almost perfectly classified all cases, with high precision, recall, and F1-score.

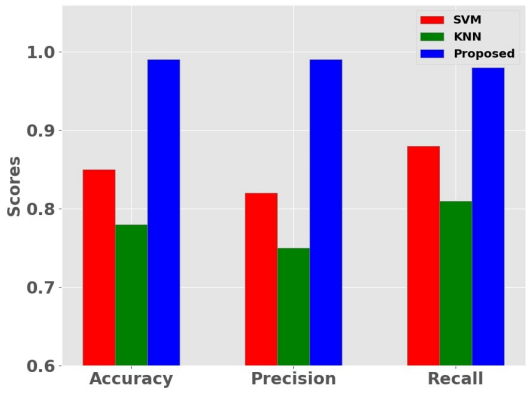

Fig. 7. Accuracy comparison of SVM, KNN, and the proposed method.

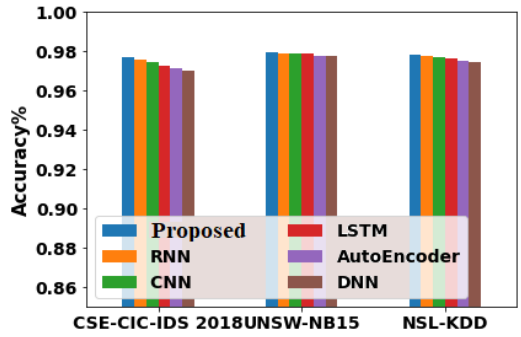

Fig. 8. Accuracy comparison between LSTM, RNN, CNN, DNN, AutoEncoder, and the proposed method.

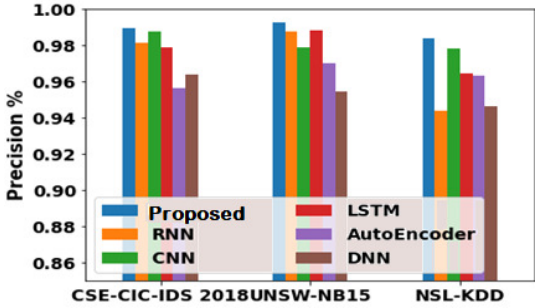

Fig. 9. Comparison of precision values among LSTM, RNN, CNN, DNN, AutoEncoder, and the proposed method.

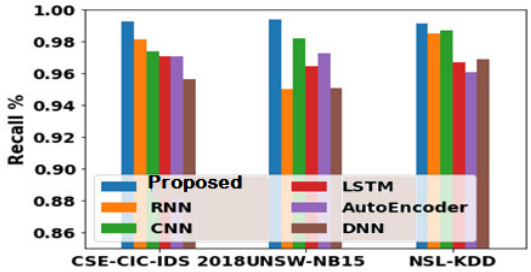

Fig. 10. Comparison of recall among LSTM, RNN, CNN, DNN, AutoEncoder, and the proposed method.

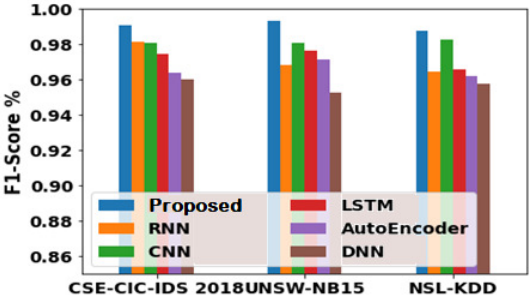

Fig. 11. Comparison of F1-score among LSTM, RNN, CNN DNN, AutoEncoder, and the proposed method.

Figure 12 compares the running time complexities (used for both training and testing) among existing DL and the proposed method. For UNSWNB-15 (with 56000 samples each for normal and attack categories after balancing) and NSL-KDD (with 67343 normal and 58630 attack samples), the running time complexity (training and testing the model) is very low at 2 s. For the CSE-CIC-IDS-2018 dataset (2,000,000 samples for each benign and attack class after balancing) the proposed method took 22 seconds to run.

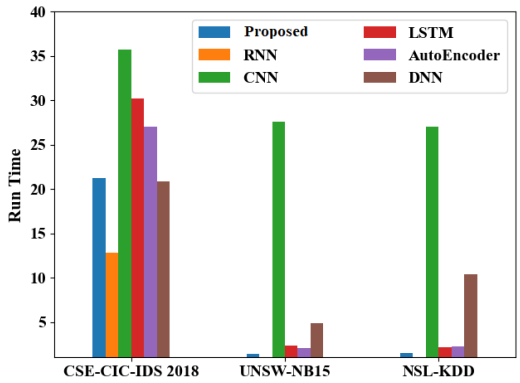

Fig. 12. Time complexities of LSTM, RNN, CNN, DNN, AutoEncoder, and the proposed method.

IV. CONCLUSION

This research on AB-SSAE for feature extraction in IDSs has shown significant promise in improving accuracy and efficiency. Through the utilization of attention mechanisms, these autoencoders can effectively identify and prioritize relevant features within network traffic data, thus improving the detection of suspicious activities and potential security breaches. The experimental results have shown notable improvements in intrusion detection performance compared to traditional methods, underscoring the potential of AB-SSAE as a valuable tool for strengthening cybersecurity defenses. The findings demonstrate that the proposed method achieved an impressive accuracy of 0.98, highlighting its exceptional ability to accurately distinguish between benign and malicious network activities and underscoring its efficacy in precisely identifying various network behaviors with high precision and reliability. Additional research efforts can explore optimizing the architecture and training strategies to address potential limitations and improve the robustness of IDSs. By continuing to innovate and refine these techniques, the cybersecurity community can leverage AB-SSAE to increase network defense resilience and mitigate evolving challenges posed by cyber threats in an increasingly interconnected digital landscape.

DECLARATIONS

A. *Conflict of Interest*

The authors declare that this research does not cause any conflict of interest.

B. *Funding*

The authors did not receive financial support or funding that could influence the findings or conclusions of this research.

REFERENCES

[1] C. Yue, L. Wang, D. Wang, R. Duo, and X. Nie, "An Ensemble Intrusion Detection Method for Train Ethernet Consist Network Based on CNN and RNN," *IEEE Access*, vol. 9, pp. 59527–59539, 2021, https://doi.org/10.1109/ACCESS.2021.3073413.

[2] J. Ghasemi, J. Esmaily, and R. Moradinezhad, "Intrusion detection system using an optimized kernel extreme learning machine and efficient features," *Sādhanā*, vol. 45, no. 1, Dec. 2019, Art. no. 2, https://doi.org/10.1007/s12046-019-1230-x.

[3] M. A. Khan and J. Kim, "Toward Developing Efficient Conv-AE-Based Intrusion Detection System Using Heterogeneous Dataset," *Electronics*, vol. 9, no. 11, Nov. 2020, Art. no. 1771, https://doi.org/10.3390/electronics9111771.

[4] P. Rajesh Kanna and P. Santhi, "Unified Deep Learning approach for Efficient Intrusion Detection System using Integrated Spatial–Temporal Features," *Knowledge-Based Systems*, vol. 226, Aug. 2021, Art. no. 107132, https://doi.org/10.1016/j.knosys.2021.107132.

[5] J. Yang, T. Li, G. Liang, W. He, and Y. Zhao, "A Simple Recurrent Unit Model Based Intrusion Detection System With DCGAN," *IEEE Access*, vol. 7, pp. 83286–83296, 2019, https://doi.org/10.1109/ACCESS.2019.2922692.

[6] T. T. H. Le, H. Kim, H. Kang, and H. Kim, "Classification and Explanation for Intrusion Detection System Based on Ensemble Trees and SHAP Method," *Sensors*, vol. 22, no. 3, Jan. 2022, Art. no. 1154, https://doi.org/10.3390/s22031154.

[7] K. H. Le, M. H. Nguyen, T. D. Tran, and N. D. Tran, "IMIDS: An Intelligent Intrusion Detection System against Cyber Threats in IoT," *Electronics*, vol. 11, no. 4, Jan. 2022, Art. no. 524, https://doi.org/10.3390/electronics11040524.

[8] V. Gowdhaman and R. Dhanapal, "An intrusion detection system for wireless sensor networks using deep neural network," *Soft Computing*, vol. 26, no. 23, pp. 13059–13067, Dec. 2022, https://doi.org/10.1007/s00500-021-06473-y.

[9] A. M. Aleesa, A. A. Mohammed, and N. M. Sahar, "Deep-intrusion detection system with enhanced UNSW-NB15 dataset based on deep learning techniques," *Journal of Engineering Science and Technology*, vol. 16, no. 1, pp. 711–727, 2021.

[10] J. Gu and S. Lu, "An effective intrusion detection approach using SVM with naïve Bayes feature embedding," *Computers & Security*, vol. 103, Apr. 2021, Art. no. 102158, https://doi.org/10.1016/j.cose.2020.102158.

[11] N. V. Sharma and N. S. Yadav, "An optimal intrusion detection system using recursive feature elimination and ensemble of classifiers," *Microprocessors and Microsystems*, vol. 85, Sep. 2021, Art. no. 104293, https://doi.org/10.1016/j.micpro.2021.104293.

[12] R. Panigrahi *et al.*, "A Consolidated Decision Tree-Based Intrusion Detection System for Binary and Multiclass Imbalanced Datasets," *Mathematics*, vol. 9, no. 7, Jan. 2021, Art. no. 751, https://doi.org/10.3390/math9070751.

[13] A. Al-Bakaa and B. Al-Musawi, "Improving the Performance of Intrusion Detection System through Finding the Most Effective Features," in *2021 International Congress of Advanced Technology and Engineering (ICOTEN)*, Taiz, Yemen, Jul. 2021, pp. 1–9, https://doi.org/10.1109/icoten52080.2021.9493564.

[14] S. Rawat, A. Srinivasan, V. Ravi, and U. Ghosh, "Intrusion detection systems using classical machine learning techniques vs integrated unsupervised feature learning and deep neural network," *Internet Technology Letters*, vol. 5, no. 1, 2022, Art. no. e232, https://doi.org/10.1002/itl2.232.

[15] M. Ramaiah, V. Chandrasekaran, V. Ravi, and N. Kumar, "An intrusion detection system using optimized deep neural network architecture," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 4, 2021, Art. no. e4221, https://doi.org/10.1002/ett.4221.

[16] S. N. Mighan and M. Kahani, "A novel scalable intrusion detection system based on deep learning," *International Journal of Information Security*, vol. 20, no. 3, pp. 387–403, Jun. 2021, https://doi.org/10.1007/s10207-020-00508-5.

[17] F. Medjek, D. Tandjaoui, N. Djedjig, and I. Romdhani, "Fault-tolerant AI-driven Intrusion Detection System for the Internet of Things," *International Journal of Critical Infrastructure Protection*, vol. 34, Sep. 2021, Art. no. 100436, https://doi.org/10.1016/j.ijcip.2021.100436.

[18] K. N. Rao, K. V. Rao, and P. V. G. D. Prasad Reddy, "A hybrid Intrusion Detection System based on Sparse autoencoder and Deep Neural Network," *Computer Communications*, vol. 180, pp. 77–88, Dec. 2021, https://doi.org/10.1016/j.comcom.2021.08.026.

[19] Y. N. Kunang, S. Nurmaini, D. Stiawan, and B. Y. Suprapto, "Attack classification of an intrusion detection system using deep learning and hyperparameter optimization," *Journal of Information Security and Applications*, vol. 58, May 2021, Art. no. 102804, https://doi.org/10.1016/j.jisa.2021.102804.

[20] S. K. Sahu, D. P. Mohapatra, J. K. Rout, K. S. Sahoo, Q. V. Pham, and N. N. Dao, "A LSTM-FCNN based multi-class intrusion detection using scalable framework," *Computers and Electrical Engineering*, vol. 99, Apr. 2022, Art. no. 107720, https://doi.org/10.1016/j.compeleceng.2022.107720.

[21] A. U. H. Qureshi, H. Larijani, M. Yousefi, A. Adeel, and N. Mtetwa, "An Adversarial Approach for Intrusion Detection Systems Using Jacobian Saliency Map Attacks (JSMA) Algorithm," *Computers*, vol. 9, no. 3, Sep. 2020, Art. no. 58, https://doi.org/10.3390/computers9030058.

[22] Y. Yang, K. Zheng, C. Wu, and Y. Yang, "Improving the Classification Effectiveness of Intrusion Detection by Using Improved Conditional Variational AutoEncoder and Deep Neural Network," *Sensors*, vol. 19, no. 11, Jan. 2019, Art. no. 2528, https://doi.org/10.3390/s19112528.

[23] S. Thakur, A. Chakraborty, R. De, N. Kumar, and R. Sarkar, "Intrusion detection in cyber-physical systems using a generic and domain specific deep autoencoder model," *Computers & Electrical Engineering*, vol. 91, May 2021, Art. no. 107044, https://doi.org/10.1016/j.compeleceng.2021.107044.

[24] A. Basati and M. M. Faghih, "APAE: an IoT intrusion detection system using asymmetric parallel auto-encoder," *Neural Computing and Applications*, vol. 35, no. 7, pp. 4813–4833, Mar. 2023, https://doi.org/10.1007/s00521-021-06011-9.

[25] F. Laghrissi, S. Douzi, K. Douzi, and B. Hssina, "Intrusion detection systems using long short-term memory (LSTM)," *Journal of Big Data*, vol. 8, no. 1, May 2021, Art. no. 65, https://doi.org/10.1186/s40537-021-00448-4.

[26] "IDS 2018 Intrusion CSVs (CSE-CIC-IDS2018)." [Online]. Available: https://www.kaggle.com/datasets/solarmainframe/ids-intrusion-csv.

[27] "UNSW_NB15." [Online]. Available: https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15.

[28] "elifnurkarakoc/CICIDS2017." [Online]. Available: https://github.com/elifnurkarakoc/CICIDS2017.