Enhancing network intrusion detection with stacked deep and reinforcement learning models.

KALPANI, N., RODRIGO, N., SENEVIRATNE, D., ARIYADASA, S. and SENANAYAKE, J.

2025

Enhancing Network Intrusion Detection with Stacked Deep and Reinforcement Learning Models

Nethma Kalpani
Department of Computer Science and Informatics
Uva Wellassa University
Badulla, Sri Lanka
cst20017@std.uwu.ac.lk

Nureka Rodrigo
Department of Computer Science and Informatics
Uva Wellassa University
Badulla, Sri Lanka
cst20069@std.uwu.ac.lk

Dilmi Seneviratne
Department of Computer Science and Informatics
Uva Wellassa University
Badulla, Sri Lanka
cst20016@std.uwu.ac.lk

Subhash Ariyadasa
Department of Computer Science and Informatics
Uva Wellassa University
Badulla, Sri Lanka
subhash@uwu.ac.lk

Janaka Senanayake
School of Computing, Engineering and Technology
Robert Gordon University
Aberdeen, AB10 7GJ, United Kingdom
j.senanayake1@rgu.ac.uk

***Abstract*—This study investigates the effectiveness of Ensemble Learning (EL) techniques by integrating reproducible Deep Learning (DL) and Reinforcement Learning (RL) models to enhance network intrusion detection. Through a systematic review of the literature, the most effective DL and RL models from 2020 to 2024 were identified based on their F1 scores and reproducibility, focusing on recent advancements in network intrusion detection. A structured normalisation and evaluation process allowed for an objective comparison of model performances. The best performing DL and RL models were subsequently integrated using a stacking ensemble technique, chosen for its ability to combine the complementary strengths of the DL and RL models. Experimental validation in a benchmark dataset confirmed the high accuracy and robust detection capabilities of the model, outperforming the individual DL and RL models to detect network intrusions in multiple classes. This research demonstrates the potential of ensemble methods for advancing Intrusion Detection Systems (IDSs), offering a scalable and effective solution for dynamic cybersecurity environments.**

***Keywords*—deep learning, ensemble learning, machine learning, network intrusion detection, reinforcement learning**

I. Introduction

The rapid expansion of the Internet and the increasing reliance on networked systems have led to a significant rise in cybersecurity threats, particularly in the form of network intrusions. As organisations become more interconnected, the need for robust IDSs has never been more critical. IDSs serve as a front-line defence mechanism, designed to monitor network traffic for suspicious activities and potential threats, thereby safeguarding sensitive information and maintaining the integrity of networks. However, traditional IDSs often struggle with high false positive rates and low detection accuracies, particularly when faced with sophisticated attacks that evolve over time [1], [2].

Recent advances in Machine Learning (ML), DL, and RL have opened new avenues to enhance the performance of IDSs. These technologies offer the potential to analyse large amounts of network data more effectively than conventional methods. Among these, EL techniques have emerged as a powerful approach to improve classification accuracy by combining multiple models to leverage their complementary strengths. By integrating various classifiers, ensemble methods can mitigate the limitations associated with individual models, such as overfitting and underperformance, in specific scenarios [3], [4]. This study explores the integration of reproducible DL and RL models within a stacking ensemble framework to enhance network intrusion detection.

II. Related Literature

Network Intrusion Detection Systems (NIDS) are vital for identifying and mitigating malicious activities within networks. Despite their effectiveness, traditional approaches face challenges in detecting novel and evolving threats, necessitating advancements in intrusion detection technologies.

Recent research highlights the potential of advanced ML techniques, particularly DL and RL, to improve NIDS. DL architectures, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have shown strong performance in handling high-dimensional and sequential data. CNNs effectively capture spatial features in network traffic, while RNNs, particularly Long-Short-Term Memory, excel at identifying temporal patterns in attacks [5], [6]. RL, offering a dynamic approach, uses techniques such as Deep Q-Networks

(DQNs) to adaptively learn from network environments, making it effective against novel and adaptive threats [7].

EL is a ML approach that enhances predictive performance by combining the outputs of multiple models. This technique leverages the strengths of various individual models, known as weak learners, to create a more robust and accurate predictive system [8]. In the context of intrusion detection, EL has emerged as a promising method for improving intrusion detection [1]. Studies by [9], [1], and [10] have explored EL frameworks, often integrating models of the same type, such as DL with DL or Conventional Machine Learning (CML) with CML. However, these approaches often fail to address the broader limitations of the DL and RL methods.

The research by [11] and [12] emphasised feature selection and EL to improve IDS performance, focusing on combinations of existing DL or CML algorithms without leveraging the complementary nature of DL and RL. Meanwhile, [13] demonstrated improved performance using a stacked EL model for wireless networks, but focused on combining gradient boosting and random forest, overlooking the advantages of DL and RL.

Despite these efforts, a significant gap remains in explicitly integrating DL and RL models within an EL framework to address their standalone limitations. This research aims to bridge this gap by leveraging the strengths of DL and RL through a stacking ensemble technique, providing a scalable and robust solution for NIDS.

III. Methodology

A. DL and RL Model Selection

A comprehensive literature review was conducted to identify effective DL and RL models for network intrusion detection, focusing on studies published between 2020 and 2024. This ensured that the findings aligned with recent advances in addressing evolving cybersecurity challenges. Key information was extracted from the selected studies, including methods, datasets, and evaluation metrics.

To allow fair comparisons, the normalised performance metrics were compiled into two tables. TABLE I for the DL models and TABLE II for the RL models, using Min-Max Scaling to ensure unbiased evaluations. The F1 score and reproducibility were prioritised as key metrics. The F1 score balances precision and recall, critical for intrusion detection with imbalanced datasets, while reproducibility ensures that models are deployable in real-world settings. The best performing models were identified based on their highest normalised scores, providing a solid foundation for the development of an effective NIDS.

The DL model from [6] and the RL model from [27] achieved the highest normalised scores, making them the best DL and RL models for this investigation.

B. EL Technique Selection

Building on the identification of the best performing DL and RL models, the next step was to develop an EL technique to combine these models.

TABLE I: EVALUATION OF IDENTIFIED DL MODELS

Study	F1 Score	Code	Dataset	Hyper parameters	Normalised Score
[14]	0.8514	0	1	0	0.044806
[15]	0.8200	0	1	1	0.250000
[16]	0.9935	0	1	0	0.247574
[17]	0.9354	0	1	0	0.164669
[18]	0.9426	0	1	1	0.424943
[19]	0.9800	0	1	1	0.478311
[20]	0.9584	0	1	1	0.447489
[21]	0.8965	0	1	1	0.359161
[22]	0.9952	0	1	1	0.500000
[23]	0.9917	1	1	0	0.495006
[24]	0.9900	0	1	1	0.492580
[25]	0.9764	0	1	0	0.245277
[6]	**0.9902**	**1**	**1**	**1**	**0.742865**
[26]	0.9946	0	1	1	0.499144

TABLE II: EVALUATION OF IDENTIFIED RL MODELS

Study	F1 Score	Code	Dataset	Hyper parameters	Normalised Score
[28]	0.9890	0	1	0	0.236224
[29]	0.9970	0	1	0	0.247029
[7]	0.8141	0	1	1	0.250000
[30]	0.9880	0	1	0	0.234873
[31]	0.9490	0	1	0	0.182199
[27]	**0.9992**	**1**	**1**	**1**	**0.750000**

Among the various ensemble methods, stacking, bagging, and boosting were evaluated. Boosting trains base learners sequentially, adjusting weights for misclassified instances, making it effective for hard-to-classify cases but sensitive to noise and outliers [32]. Bagging, on the other hand, is a parallel method that reduces variance by training multiple base learners on bootstrapped data subsets, which performs well with high-variance models such as Decision Trees (DTs), but is less effective for weak learners to address bias [32].

Stacking is especially beneficial when integrating diverse models with complementary strengths, making it an ideal choice for complex tasks such as network intrusion detection. Unlike boosting and bagging, stacking is versatile and can handle models with varying characteristics, allowing each model to contribute its unique insights to a unified prediction [32].

Therefore, stacking was identified as the optimal EL technique for integrating the selected DL and RL models, offering a powerful way to improve the performance of NIDSs.

C. Metaclassifier Selection

The next step involved selecting suitable metaclassifiers for the ensemble model. A comprehensive literature review was conducted to identify ensemble techniques commonly used in network intrusion detection. This informed the selection of metaclassifiers for the initial ensemble models.

Hossain et al. [33] proposed an EL technique for intrusion detection, comparing several methods such as Random Forest (RF), Gradient Boosting, AdaBoost, and XGBoost. Their study

found that RF outperformed others in terms of accuracy and false positive rates, achieving high precision, recall, F1 score, and balanced accuracy. Based on these findings, RF was chosen as one of the metaclassifiers.

Thockchom et al. [1] introduced a stacking-based model using Gaussian Naive Bayes (GNB), DT, and Logistic Regression (LR) as base classifiers, with stochastic gradient descent as metaclassifier. Their approach achieved an accuracy of 99.48%, motivating the development of similar stacking models for this investigation.

Ali et al. [34] proposed another stacking ensemble model using k-Nearest neighbors, support vector machine, and RF with XGBoost as meta-classifier, achieving a weighted F1 score of 98.24%. This inspired the implementation of an additional ensemble model using XGBoost as the metaclassifier.

In addition, other classifiers such as Histogram Gradient Boosting (HGB), Linear Discriminant Analysis (LDA), and Quadratic Discriminant Analysis (QDA) were explored to assess their effectiveness within this study.

D. Reproduction of Selected DL Model

Building on the previous steps, the DL model chosen for reproduction is from [6], which uses the CSE-CIC-IDS2018 [35] dataset, the same dataset used in this investigation as the benchmark dataset.

The reproduction process followed the steps outlined in the original study, including data preprocessing, model implementation, and training configurations. Key settings such as early stopping with a patience of 20 and a total of 500 epochs were used to prevent overfitting. After executing the training process, the performance metrics achieved were compared with those of the original study. The reproduction resulted in an accuracy of 0.9913 and a loss of 0.0041, which were slightly higher than the accuracy reported by the original study of 0.9910 and a loss of 0.0040.

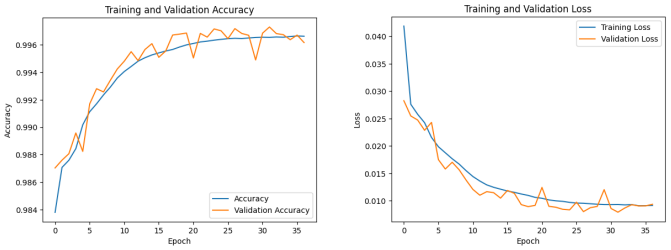
The training and validation performance curves are shown in Fig. 1.

It should be noted that while [6] excluded certain attack classes (DDoS attacks-LOIC-HTTP, FTP-BruteForce, and SSH-BruteForce), this research included all 15 attack classes from the CSE-CIC-IDS2018 dataset to maintain consistency with the RL models. Despite this difference, the reproduction still resulted in slightly better accuracy, suggesting that including additional attack classes did not negatively impact the model's performance.

E. Reproduction of Selected RL Model

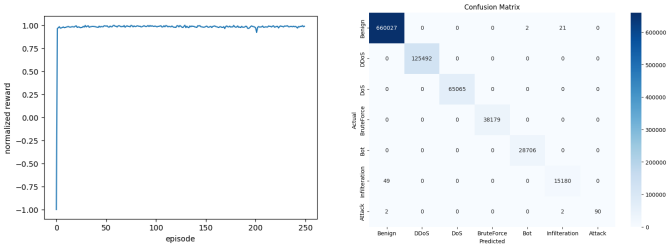
The next step of the investigation focused on reproducing the selected RL model presented by [27], which also used the CSE-CIC-IDS2018 dataset. The reproduction process followed the steps outlined in the original study, including data preprocessing, model implementation, and training configurations.

The RL model in [27] involves training an agent over 250 episodes, each episode consisting of 200 iterations, resulting in a total of 50,000 interactions between the agent and the environment. This comprehensive training process allowed the



(a) Training and validation accuracy (b) Training and validation loss

Fig. 1: Performance of DL model



(a) Normalised reward per episode (b) Confusion matrix

Fig. 2: Performance of RL model

RL agent to develop robust detection capabilities and adapt to the diverse characteristics of the dataset.

After training, the reproduction achieved an accuracy of 0.9997, slightly higher than the accuracy reported in the original study of 0.9992. This confirmed the successful reproduction of the RL model, with performance closely matching and even improving on the original results.

A notable difference was observed in the classification performance for the 'Attack' class. In the original study, the model reported an accuracy of 0 for the 'Attack' class. In contrast, the reproduced model achieved an accuracy of 0.9245 for this class, while maintaining slightly better accuracies for other classes. This improvement indicates that the reproduced RL model was more effective in identifying attack instances compared to the original study.

The training and validation performance of the RL model is shown in Fig. 2.

IV. Results

The ensemble models were developed by feeding the predictions from the selected DL and RL models into the selected meta-classifiers. The results of each metaclassifier are displayed in Table III-Table X and Fig. 3-Fig. 10.

The evaluation of the ensemble models developed was based on several key metrics: precision, recall, F1 score, accuracy, and inference time. Each metric was carefully analysed to assess the overall effectiveness and computational feasibility of the ensemble models in real-world applications.

A performance overview for all metaclassifiers tested was summarised in the TABLE XI, highlighting the comparative results across these metrics.

Among the tested classifiers, the HGB meta-classifier demonstrated the highest balanced accuracy (0.9951), making

it the most effective choice for this ensemble framework. RF also performed well, achieving similar precision and recall values, but with slightly longer inference time. In contrast, classifiers such as GNB and QDA showed lower accuracy and recall, which could be attributed to their inability to capture the complexity of the ensemble's input data distribution.

The models were also compared for computational efficiency, a crucial aspect for IDSs that require real-time or near-real-time processing. HGB not only provided robust classification performance, but also balanced computational cost with an inference time of 4.54 seconds per batch, outperforming the RF's inference time of 11.09 seconds. These findings indicate that HGB strikes an optimal balance between accuracy and speed, making it suitable for deployment in dynamic network environments.

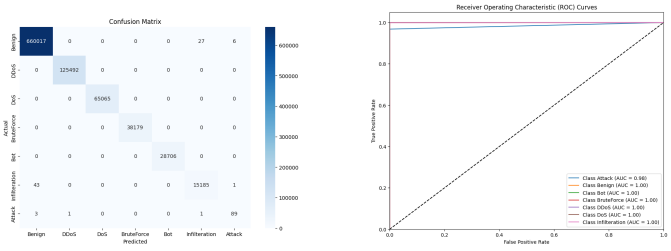
V. Discussion

The results underscore the effectiveness of the stacking ensemble method for intrusion detection. By combining predictions from the DL and RL models, the ensemble addressed individual model limitations and leveraged their complementary strengths. The DL model excelled at detecting majority classes, while the RL model effectively identified minority classes, resulting in improved classification accuracy and reduced false positives and negatives. This approach surpasses previous work that relied on standalone DL or RL models. The ensemble framework, with robust metaclassifiers like HGB, demonstrated superior adaptability and scalability, generalising well across diverse attack types in imbalanced datasets.

However, the ensemble method has challenges, including increased computational complexity from training multiple base models and a meta-classifier. This can be mitigated through optimised training pipelines or distributed computing. Furthermore, while the proposed approach performed well on

TABLE III: MULTICLASS CLASSIFICATION OF DT

Label	Precision	Recall	F1 Score	AUC Score	Support
Benign	1.00	1.00	1.00	1.00	660050
DDos	1.00	1.00	1.00	1.00	125492
Dos	1.00	1.00	1.00	1.00	65065
BruteForce	1.00	1.00	1.00	1.00	38179
Bot	1.00	1.00	1.00	1.00	28706
Infilteration	1.00	1.00	1.00	1.00	15229
Attack	0.93	0.95	0.94	0.98	94

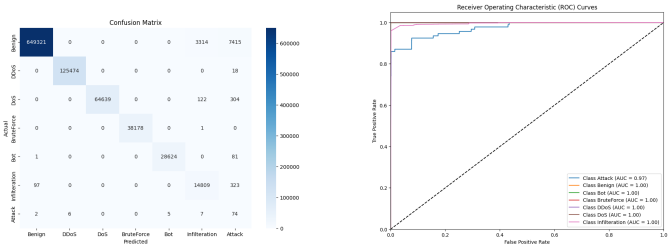


(a) Confusion matrix (b) ROC curve

Fig. 3: Performance of DT

TABLE IV: MULTICLASS CLASSIFICATION OF GNB

Label	Precision	Recall	F1 Score	AUC Score	Support
Benign	1.00	0.98	0.99	1.00	660050
DDos	1.00	1.00	1.00	1.00	125492
Dos	1.00	1.00	1.00	1.00	65065
BruteForce	1.00	1.00	1.00	1.00	38179
Bot	1.00	1.00	1.00	1.00	28706
Infilteration	0.81	0.97	0.88	1.00	15229
Attack	0.01	0.79	0.02	0.97	94

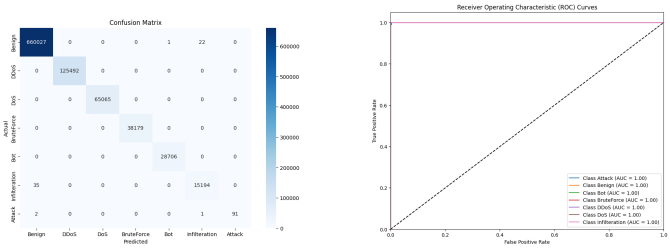


(a) Confusion matrix (b) ROC curve

Fig. 4: Performance of GNB

TABLE V: MULTICLASS CLASSIFICATION OF HGB

Label	Precision	Recall	F1 Score	AUC Score	Support
Benign	1.00	1.00	1.00	1.00	660050
DDos	1.00	1.00	1.00	1.00	125492
Dos	1.00	1.00	1.00	1.00	65065
BruteForce	1.00	1.00	1.00	1.00	38179
Bot	1.00	1.00	1.00	1.00	28706
Infilteration	1.00	1.00	1.00	1.00	15229
Attack	1.00	0.97	0.98	1.00	94

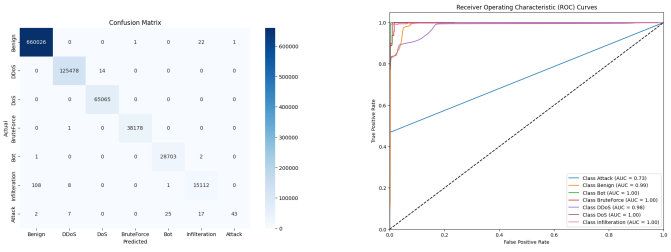


(a) Confusion matrix (b) ROC curve

Fig. 5: Performance of HGB

TABLE VI: MULTICLASS CLASSIFICATION OF LDA

Label	Precision	Recall	F1 Score	AUC Score	Support
Benign	1.00	1.00	1.00	0.99	660050
DDos	1.00	1.00	1.00	0.98	125492
Dos	1.00	1.00	1.00	1.00	65065
BruteForce	1.00	1.00	1.00	1.00	38179
Bot	1.00	1.00	1.00	1.00	28706
Infilteration	1.00	0.99	0.99	1.00	15229
Attack	0.98	0.46	0.62	0.73	94

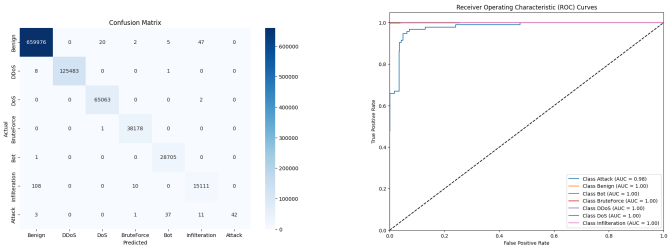


(a) Confusion matrix (b) ROC curve

Fig. 6: Performance of LDA

TABLE VII: MULTICLASS CLASSIFICATION OF LR

Label	Precision	Recall	F1 Score	AUC Score	Support
Benign	1.00	1.00	1.00	1.00	660050
DDos	1.00	1.00	1.00	1.00	125492
Dos	1.00	1.00	1.00	1.00	65065
BruteForce	1.00	1.00	1.00	1.00	38179
Bot	1.00	1.00	1.00	1.00	28706
Infilteration	1.00	0.99	0.99	1.00	15229
Attack	1.00	0.45	0.62	0.98	94

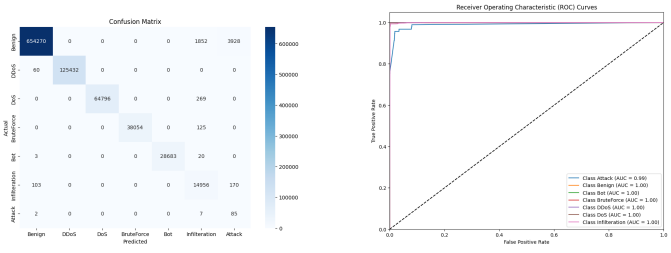


(a) Confusion matrix (b) ROC curve

Fig. 7: Performance of LR

TABLE VIII: MULTICLASS CLASSIFICATION OF QDA

Label	Precision	Recall	F1 Score	AUC Score	Support
Benign	1.00	0.99	1.00	1.00	660050
DDos	1.00	1.00	1.00	1.00	125492
Dos	1.00	1.00	1.00	1.00	65065
BruteForce	1.00	1.00	1.00	1.00	38179
Bot	1.00	1.00	1.00	1.00	28706
Infilteration	0.87	0.98	0.92	1.00	15229
Attack	0.02	0.90	0.04	0.99	94

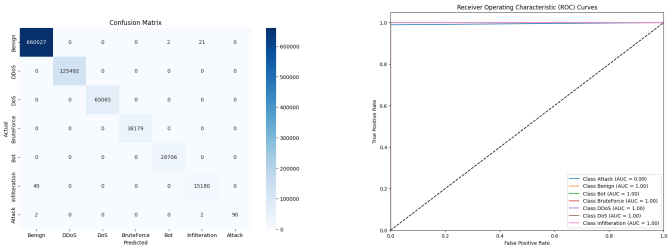


(a) Confusion matrix (b) ROC curve

Fig. 8: Performance of QDA

TABLE IX: MULTICLASS CLASSIFICATION OF RF

Label	Precision	Recall	F1 Score	AUC Score	Support
Benign	1.00	1.00	1.00	1.00	660050
DDos	1.00	1.00	1.00	1.00	125492
Dos	1.00	1.00	1.00	1.00	65065
BruteForce	1.00	1.00	1.00	1.00	38179
Bot	1.00	1.00	1.00	1.00	28706
Infilteration	1.00	1.00	1.00	1.00	15229
Attack	1.00	0.96	0.98	0.99	94

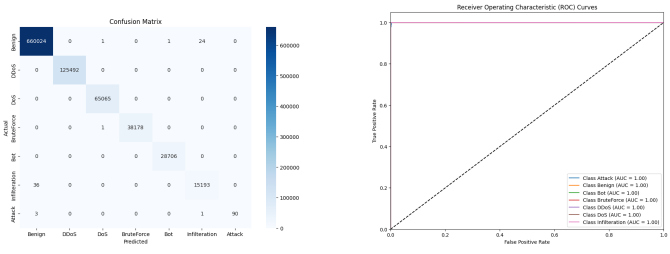


(a) Confusion matrix (b) ROC curve

Fig. 9: Performance of RF

TABLE X: MULTICLASS CLASSIFICATION OF XGBoost

Label	Precision	Recall	F1 Score	AUC Score	Support
Benign	1.00	1.00	1.00	1.00	660050
DDos	1.00	1.00	1.00	1.00	125492
Dos	1.00	1.00	1.00	1.00	65065
BruteForce	1.00	1.00	1.00	1.00	38179
Bot	1.00	1.00	1.00	1.00	28706
Infilteration	1.00	1.00	1.00	1.00	15229
Attack	1.00	0.96	0.98	1.00	94



(a) Confusion matrix (b) ROC curve

Fig. 10: Performance of XGBoost

TABLE XI: SUMMARY OF CLASSIFIER PERFORMANCE

Classifier	Accuracy	Precision	Recall	F1 Score	Balanced Accuracy	Inference Time
DT	0.9999	0.9999	0.9999	0.9999	0.9920	0.28s
GNB	0.9875	0.9967	0.9875	0.9919	0.9620	5.92s
HGB	**0.9999**	**0.9999**	**0.9999**	**0.9999**	**0.9951**	**4.54s**
LDA	0.9998	0.9998	0.9998	0.9998	0.9214	1.98s
LR	0.9997	0.9997	0.9997	0.9997	0.9198	0.63s
QDA	0.9930	0.9976	0.9930	0.9952	0.9813	2.15s
RF	0.9999	0.9999	0.9999	0.9999	0.9935	11.09s
XGBoost	0.9999	0.9999	0.9999	0.9999	0.9936	1.21s

the benchmark CSE-CIC-IDS2018 dataset, its effectiveness with live traffic data and evolving attack patterns requires further validation.

In general, this study highlights the potential of ensemble methods in advancing intrusion detection systems. By combining accuracy with scalability, the proposed approach provides a strong foundation for resilient and efficient cybersecurity solutions in dynamic network environments.

VI. Conclusion

This research highlights the transformative potential of EL techniques in advancing NIDSs. By systematically integrating state-of-the-art DL and RL models within a stacking framework, the study achieved significant improvements in detection accuracy, particularly across various attack classes. The implementation of an optimised metaclassifier, such as HGB, was instrumental in striking an effective balance between detection performance and computational efficiency, making the proposed approach suitable for scalable and real-time cybersecurity applications.

Furthermore, the research underscores the importance of reproducibility and dataset diversity in the development and evaluation of robust intrusion detection models. By including all attack classes in the CSE-CIC-IDS2018 dataset, the study addressed critical gaps in previous work, ensuring the adaptability of the proposed model to real-world scenarios. This comprehensive approach not only improves the model's robustness, but also enhances its generalisability to complex and evolving network environments.

Future research could explore the integration of additional features, such as temporal data patterns, contextual analysis, and user behaviour monitoring, to further improve detection accuracy. Furthermore, expanding the framework to include adversarial training techniques could enhance its resilience against attempts to deceive the system. Real-time implementation and validation in operational environments with live traffic data would also be critical steps to bridge the gap between theoretical advances and practical deployment. The proposed approach sets the foundation for developing resilient and scalable cybersecurity systems capable of tackling ever-evolving threats in dynamic network environments.

References

[1] N. Thockchom, M. M. Singh, and U. Nandi, "A Novel Ensemble Learning-based Model for Network Intrusion Detection," *Complex & Intelligent Systems*, vol. 9, no. 5, pp. 5693–5714, Apr. 2023.

[2] I. P. J. Vijaya K. Shandilya, "A Critical Review - Use of Ensemble Methods in Intrusion Detection System," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12, no. 21s, p. 1157–1164, Mar. 2024.

[3] A. M. Alsaffar, M. Nouri-Baygi, and H. M. Zolbanin, "Shielding networks: enhancing intrusion detection with hybrid feature selection and stack ensemble learning," *Journal of Big Data*, vol. 11, no. 1, Sep. 2024.

[4] Y. Alotaibi and M. Ilyas, "Ensemble-Learning Framework for Intrusion Detection to Enhance Internet of Things' Devices Security," *Sensors*, vol. 23, no. 12, p. 5568, Jun. 2023.

[5] W. Wang, M. Zhu, X. Zeng, X. Ye, and Y. Sheng, "Malware Traffic Classification Using Convolutional Neural Network for Representation Learning," in *2017 International Conference on Information Networking (ICOIN)*. IEEE, 2017.

[6] V. Hnamte, H. Nhung-Nguyen, J. Hussain, and Y. Hwa-Kim, "A Novel Two-Stage Deep Learning Model for Network Intrusion Detection: LSTM-AE," *IEEE Access*, vol. 11, pp. 37 131–37 148, 2023.

[7] H. Alavizadeh, H. Alavizadeh, and J. Jang-Jaccard, "Deep Q-Learning Based Reinforcement Learning Approach for Network Intrusion Detection," *Computers*, vol. 11, no. 3, p. 41, Mar. 2022.

[8] F. Huang, G. Xie, and R. Xiao, "Research on Ensemble Learning," in *2009 International Conference on Artificial Intelligence and Computational Intelligence*. IEEE, 2009.

[9] V. Dutta, M. Choraś, M. Pawlicki, and R. Kozik, "A Deep Learning Ensemble for Network Anomaly and Cyber-Attack Detection," *Sensors*, vol. 20, no. 16, p. 4583, Aug. 2020.

[10] Imran, F. Jamil, and D. Kim, "An Ensemble of Prediction and Learning Mechanism for Improving Accuracy of Anomaly Detection in Network Intrusion Environments," *Sustainability*, vol. 13, no. 18, p. 10057, Sep. 2021.

[11] Q. R. S. Fitni and K. Ramli, "Implementation of Ensemble Learning and Feature Selection for Performance Improvements in Anomaly-Based Intrusion Detection Systems," in *2020 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)*. IEEE, Jul. 2020, pp. 118–124.

[12] X. Gao, C. Shan, C. Hu, Z. Niu, and Z. Liu, "An Adaptive Ensemble Machine Learning Model for Intrusion Detection," *IEEE Access*, vol. 7, p. 82512–82521, 2019.

[13] H. Rajadurai and U. D. Gandhi, "A stacked ensemble learning model for intrusion detection in wireless network," *Neural Computing and Applications*, vol. 34, no. 18, p. 15387–15395, May 2020.

[14] K. Jiang, W. Wang, A. Wang, and H. Wu, "Network Intrusion Detection Combined Hybrid Sampling With Deep Hierarchical Network," *IEEE Access*, vol. 8, p. 32464–32476, 2020.

[15] A. Kim, M. Park, and D. H. Lee, "AI-IDS: Application of Deep Learning to Real-Time Web Intrusion Detection," *IEEE Access*, vol. 8, p. 70245–70261, 2020.

[16] J. Malik, A. Akhunzada, I. Bibi, M. Imran, A. Musaddiq, and S. W. Kim, "Hybrid Deep Learning: An Efficient Reconnaissance and Surveillance Detection Mechanism in SDN," *IEEE Access*, vol. 8, p. 134695–134706, 2020.

[17] Y. Yang, K. Zheng, B. Wu, Y. Yang, and X. Wang, "Network Intrusion Detection Based on Supervised Adversarial Variational Auto-Encoder With Regularization," *IEEE Access*, vol. 8, p. 42169–42184, 2020.

[18] Y. Imrana, Y. Xiang, L. Ali, and Z. Abdul-Rauf, "A bidirectional LSTM deep learning approach for intrusion detection," *Expert Systems with Applications*, vol. 185, p. 115524, Dec. 2021.

[19] R. V. Mendonca, A. A. M. Teodoro, R. L. Rosa, M. Saadi, D. C. Melgarejo, P. H. J. Nardelli, and D. Z. Rodriguez, "Intrusion Detection System Based on Fast Hierarchical Deep Convolutional Neural Network," *IEEE Access*, vol. 9, p. 61024–61034, 2021.

[20] Z. Wang, Y. Liu, D. He, and S. Chan, "Intrusion detection methods based on integrated deep learning model," *Computers & Security*, vol. 103, p. 102177, Apr. 2021.

[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, p. 898, Mar. 2022.

[22] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "CNN-LSTM: Hybrid Deep Neural Network for Network Intrusion Detection System," *IEEE Access*, vol. 10, p. 99837–99849, 2022.

[23] Z. Wu, H. Zhang, P. Wang, and Z. Sun, "RTIDS: A Robust Transformer-Based Approach for Intrusion Detection System," *IEEE Access*, vol. 10, p. 64375–64387, 2022.

[24] O. Elnakib, E. Shaaban, M. Mahmoud, and K. Emara, "EIDM: deep learning model for IoT intrusion detection systems," *The Journal of Supercomputing*, vol. 79, no. 12, p. 13241–13261, Mar. 2023.

[25] J. Figueiredo, C. Serrão, and A. M. de Almeida, "Deep Learning Model Transposition for Network Intrusion Detection Systems," *Electronics*, vol. 12, no. 2, p. 293, Jan. 2023.

[26] M. B. Umair, Z. Iqbal, M. A. Faraz, M. A. Khan, Y.-D. Zhang, N. Razmjooy, and S. Kadry, "A Network Intrusion Detection System Using Hybrid Multilayer Deep Learning Model," *Big Data*, vol. 12, no. 5, p. 367–376, Oct. 2024.

[27] B. Darabi, M. Bag-Mohammadi, and M. Karami, "A micro Reinforcement Learning architecture for Intrusion Detection Systems," *Pattern Recognition Letters*, vol. 185, p. 81–86, Sep. 2024.

[28] K. Sethi, R. Kumar, N. Prajapati, and P. Bera, "Deep Reinforcement Learning based Intrusion Detection System for Cloud Infrastructure," in *2020 International Conference on COMmunication Systems & NETworkS (COMSNETS)*. IEEE, Jan. 2020.

[29] M. Lopez-Martin, A. Sanchez-Esguevillas, J. I. Arribas, and B. Carro, "Network Intrusion Detection Based on Extended RBF Neural Network With Offline Reinforcement Learning," *IEEE Access*, vol. 9, p. 153153–153170, 2021.

[30] V. Praveena, A. Vijayaraj, P. Chinnasamy, I. Ali, R. Alroobaea, S. Yahya Alyahyan, and M. Ahsan Raza, "Optimal Deep Reinforcement Learning for Intrusion Detection in UAVs," *Computers, Materials & Continua*, vol. 70, no. 2, p. 2639–2653, 2022.

[31] K. Ren, Y. Zeng, Z. Cao, and Y. Zhang, "ID-RDRL: a deep reinforcement learning-based feature selection intrusion detection model," *Scientific Reports*, vol. 12, no. 1, Sep. 2022.

[32] T. N. Rincy and R. Gupta, "Ensemble Learning Techniques and its Efficiency in Machine Learning: A Survey," in *2nd International Conference on Data, Engineering and Applications (IDEA)*. IEEE, Feb. 2020.

[33] M. A. Hossain and M. S. Islam, "Ensuring network security with a robust intrusion detection system using ensemble-based machine learning," *Array*, vol. 19, p. 100306, Sep. 2023.

[34] M. Ali, M.-u. Haque, M. H. Durad, A. Usman, S. M. Mohsin, H. Mujlid, and C. Maple, "Effective network intrusion detection using stacking-based ensemble approach," *International Journal of Information Security*, vol. 22, no. 6, p. 1781–1798, Jul. 2023.

[35] Canadian Institute for Cybersecurity (CIC), "A Realistic Cyber Defense Dataset (CSE-CIC-IDS2018)," https://registry.opendata.aws/cse-cic-ids2018, 2018, accessed: 2024-10-9.