REFERENCES

Aamir, M. and Zaidi, S.M.A., (2019). DDoS attack detection with feature engineering and machine learning: the framework and performance evaluation. *International Journal of Information Security*, 18, pp.761-785.

Abdullah Emir Cil and Kazim Yildiz and Ali Buldu, (2021). Detection of DDoS attacks with feed forward based deep neural network model. Expert Systems with Applications, 169, pp. 114520

Adams, S. and Beling, P.A., (2019). A survey of feature selection methods for Gaussian mixture models and hidden Markov models. Artificial Intelligence Review, 52, pp.1739-1779.

Admass, W.S., Munaye, Y.Y. and Diro, A.A., (2024). Cyber security: State of the art, challenges and future directions. Cyber Security and Applications, 2, p.100031.

Aguru, Aswani Devi, and Suresh Babu Erukala. (2024). A lightweight multi-vector DDoS detection framework for IoT-enabled mobile health informatics systems using deep learning. Information Sciences 662, pp.120209.

Ahmetoglu, H. and Das, R., (2022). A comprehensive review on detection of cyber-attacks: Data sets, methods, challenges, and future research directions. Internet of Things, 20, p.100615.

Ahn S, Yi H, Lee Y, Ha W R, Kim G and Paek Y. (2020). Hawkware: Network Intrusion Detection based on Behavior Analysis with ANNs on an IoT Device. 57th ACM/IEEE Design Automation Conference (DAC), San Francisco, CA, USA. pp. 1-6

Akgun, Devrim, Selman Hizal, and Unal Cavusoglu. (2022). A new DDoS attacks intrusion detection model based on deep learning for cybersecurity. Computers & Security. 118 pp.102748.

Alabdulwahab, Saleh, and BongKyo Moon. (2020). Feature selection methods simultaneously improve the detection accuracy and model building time of machine learning classifiers. Symmetry 12.9 pp.1424.

Alghazzawi, D., Bamasag, O., Ullah, H. and Asghar, M.Z., (2021). Efficient detection of DDoS attacks using a hybrid deep learning model with improved feature selection. Applied Sciences. 11(24) pp.11634.

Aljawarneh, Shadi, Monther Aldwairi, and Muneer Bani Yassein. (2018). Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. Journal of Computational Science 25 pp.152-160.

Al-Omari, M., Rawashdeh, M., Qutaishat, F., Alshira'H, M. and Ababneh, N., (2021). An intelligent tree-based intrusion detection model for cyber security. Journal of Network and Systems Management, 29(2), p.20.

Amiri, F., Yousefi, M.R., Lucas, C., Shakery, A. and Yazdani, N., (2011). Mutual information-based feature selection for intrusion detection systems. Journal of Network and Computer Applications, 34(4), pp.1184-1199.

Baig, Z.A., Sait, S.M. and Shaheen, A., (2013). GMDH-based networks for intelligent intrusion detection. Engineering Applications of Artificial Intelligence, 26(7), pp.1731-1740.

Batchu, Raj Kumar, and Hari Seetha. (2022). On improving the performance of DDoS attack detection system. Microprocessors and Microsystems 93 pp.104571.

Bedi, P., Gupta, N. and Jindal, V., (2021). I-SiamIDS: an improved Siam-IDS for handling class imbalance in network-based intrusion detection systems. Applied Intelligence, 51(2), pp.1133-1151.

Behal, S. and Kumar, K., (2016). Trends in validation of DDoS research. Procedia Computer Science, 85, pp.7-15.

Bharati, M. and Tamane, S., (2017), October. Intrusion detection systems (IDS) & future challenges in cloud based environment. In 2017 1st International Conference on Intelligent Systems and Information Management (ICISIM) (pp. 240-250). IEEE.

Bhardwaj, A., Mangat, V., Vig, R., Halder, S. and Conti, M., (2021). Distributed denial of service attacks in cloud: State-of-the-art of scientific and commercial solutions. Computer Science Review, 39, p.100332.

Bhol, S.G., Mohanty, J.R. and Pattnaik, P.K., (2023). Taxonomy of cyber security metrics to measure strength of cyber security. Materials Today: Proceedings, 80, pp.2274-2279.

Bi, J., Xu, K., Yuan, H., Zhang, J. and Zhou, M., (2023). Network attack prediction with hybrid temporal convolutional network and bi-directional GRU. IEEE Internet of Things Journal.

Boothalingam, R., (2018). Optimization using lion algorithm: a biological inspiration from lion's social behavior. Evolutionary Intelligence, 11(1), pp.31-52.

Bouke, M.A., Abdullah, A., ALshatebi, S.H., Abdullah, M.T. and El Atigh, H., (2023). An intelligent DDoS attack detection tree-based model using Gini index feature selection method. Microprocessors and Microsystems, 98, p.104823.

Brahma, K.K., Sarmah, S., Kalita, C. and Ghosh, R., (2019). Detection of Multi-Vector DDoS Attack. Int. J. Comput. Sci. Eng, 7(6), pp.847-851.

Chivukula, R., Lakshmi, T.J., Kandula, L.R.R. and Alla, K., (2021), November. A study of cyber security issues and challenges. In 2021 IEEE Bombay Section Signature Conference (IBSSC) (pp. 1-5). IEEE.

Chung, J., Gulcehre, C., Cho, K. and Bengio, Y., (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555.

Cil, A.E., Yildiz, K. and Buldu, A., (2021). Detection of DDoS attacks with feed forward based deep neural network model. Expert Systems with Applications, 169, p.114520.

Das, S., Venugopal, D., Shiva, S. and Sheldon, F.T., (2020), August. Empirical evaluation of the ensemble framework for feature selection in ddos attack. In 2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom) (pp. 56-61). IEEE.

de Neira, A.B., Kantarci, B. and Nogueira, M., (2023). Distributed denial of service attack prediction: Challenges, open issues and opportunities. Computer Networks, 222, p.109553.

De Sousa, M.S., Veiga, C.E.L., Albuquerque, R.D.O. and Giozza, W.F., (2022), June. Information Gain applied to reduce model-building time in decision-tree-based intrusion detection system. In 2022 17th Iberian Conference on Information Systems and Technologies (CISTI) (pp. 1-6). IEEE.

Devan, P. and Khare, N., (2020). An efficient XGBoost–DNN-based classification model for network intrusion detection system. Neural Computing and Applications, 32(16), pp.12499-12514.

Devi, K.L., Subathra, P. and Kumar, P.N., (2015). Tweet sentiment classification using an ensemble of machine learning supervised classifiers employing statistical feature selection methods. In Proceedings of the Fifth International Conference on Fuzzy and Neuro Computing (FANCCO-2015) (pp. 1-13). Springer International Publishing.

Du, D., Zhu, M., Li, X., Fei, M., Bu, S., Wu, L. and Li, K., (2022). A review on cybersecurity analysis, attack detection, and attack defense methods in cyber-physical power systems. Journal of Modern Power Systems and Clean Energy, 11(3), pp.727-743.

Duo, W., Zhou, M. and Abusorrah, A., (2022). A survey of cyber attacks on cyber physical systems: Recent advances and challenges. IEEE/CAA Journal of Automatica Sinica, 9(5), pp.784-800.

Elghazel, H. and Aussem, A., (2015). Unsupervised feature selection with ensemble learning. Machine Learning, 98, pp.157-180.

Eliyan, L.F. and Di Pietro, R., (2021). DoS and DDoS attacks in Software Defined Networks: A survey of existing solutions and research challenges. Future Generation Computer Systems, 122, pp.149-171.

Elsayed, M.A. and Zulkernine, M., (2020). PredictDeep: security analytics as a service for anomaly detection and prediction. IEEE Access, 8, pp.45184-45197.

Ferhi, W., Moussaoui, D., Hadjila, M., Bouidaine, A. and Bensenouci, D., (2023). DDoS Attacks Detection and Classification based on Deep Learning Model.

Garg, S., Kaur, K., Kumar, N., Kaddoum, G., Zomaya, A.Y. and Ranjan, R., (2019). A hybrid deep learning-based model for anomaly detection in cloud datacenter

networks. IEEE Transactions on Network and Service Management, 16(3), pp.924-935.

Gaurav, A., Gupta, B.B., Alhalabi, W., Visvizi, A. and Asiri, Y., (2022). A comprehensive survey on DDoS attacks on various intelligent systems and it's defense techniques. International Journal of Intelligent Systems, 37(12), pp.11407-11431.

Ghanem, W.A. and Jantan, A., (2020). A new approach for intrusion detection system based on training multilayer perceptron by using enhanced Bat algorithm. Neural Computing and Applications, 32(15), pp.11665-11698.

Goodfellow, I., Bengio, Y. and Courville, A., (2016). Deep feedforward networks. Deep learning, (1).

Gu, J. and Lu, S., (2021). An effective intrusion detection approach using SVM with naïve Bayes feature embedding. Computers & Security, 103, p.102158.

Gupta, B.B. and Dahiya, A., (2021). Distributed Denial of Service (DDoS) Attacks: Classification, Attacks, Challenges and Countermeasures. CRC press.

Gupta, N., Jindal, V. and Bedi, P., (2021). LIO-IDS: Handling class imbalance using LSTM and improved one-vs-one technique in intrusion detection system. Computer Networks, 192, p.108076.

Hajimirzaei, B. and Navimipour, N.J., (2019). Intrusion detection for cloud computing using neural networks and artificial bee colony optimization algorithm. Ict Express, 5(1), pp.56-59.

Hamarshe, A., Ashqar, H.I. and Hamarsheh, M., (2023), May. Detection of DDoS Attacks in Software Defined Networking Using Machine Learning Models. In International Conference on Advances in Computing Research (pp. 640-651). Cham: Springer Nature Switzerland.

Ho, C.Y., Lai, Y.C., Chen, I.W., Wang, F.Y. and Tai, W.H., (2012). Statistical analysis of false positives and false negatives from real traffic with intrusion detection/prevention systems. IEEE Communications Magazine, 50(3), pp.146-154.

Husák, M., Komárková, J., Bou-Harb, E. and Čeleda, P., (2019). Survey of attack projection, prediction, and forecasting in cyber security. IEEE Communications Surveys & Tutorials, 21(1), pp.640-660.

Hussain, Y.S., (2020). Network Intrusion Detection for Distributed Denial-of-Service (DDoS) Attacks using Machine Learning Classification Techniques. dspace.library.uvic.ca

Ibrahim, Z.K. and Thanon, M.Y., (2021), January. Performance comparison of intrusion detection system using three different machine learning algorithms. In 2021 6th international conference on inventive computation technologies (ICICT) (pp. 1116-1124). IEEE.

Jha, S., Prashar, D., Long, H.V. and Taniar, D., (2020). Recurrent neural network for detecting malware. computers & security, 99, p.102037.

John, J. and Norman, J., (2019). Major vulnerabilities and their prevention methods in cloud computing. In Advances in Big Data and Cloud Computing: Proceedings of ICBDCC18 (pp. 11-26). Springer Singapore.

Jose, A.S., Nair, L.R. and Paul, V., (2021). Towards detecting flooding DDOS attacks over software defined networks using machine learning techniques. Revista Geintec-Gestao Inovacao E Tecnologias, 11(4), pp.3837-3865.

Jyothsna, V.V.R.P.V., Prasad, R. and Prasad, K.M., (2011). A review of anomaly based intrusion detection systems. International Journal of Computer Applications, 28(7), pp.26-35.

Kadri, M.R., Abdelli, A., Othman, J.B. and Mokdad, L., (2024). Survey and classification of Dos and DDos attack detection and validation approaches for IoT environments. Internet of Things, 25, p.101021.

Karatas, G. and Sahingoz, O.K., (2018), March. Neural network based intrusion detection systems with different training functions. In 2018 6th International Symposium on Digital Forensic and Security (ISDFS) (pp. 1-6). IEEE.

Kasim, Ö., (2020). An efficient and robust deep learning based network anomaly detection against distributed denial of service attacks. Computer Networks, 180, p.107390.

Kasongo, S.M. and Sun, Y., (2019). A deep learning method with filter based feature engineering for wireless intrusion detection system. IEEE access, 7, pp.38597-38607.

Kaur, J. and Ramkumar, K.R., (2022). The recent trends in cyber security: A review. Journal of King Saud University-Computer and Information Sciences, 34(8), pp.5766-5781.

Keller, A., Borkmann, D., Neuhaus, S. and Happe, M., (2014). Self- Awareness in Computer Networks. International Journal of Reconfigurable Computing, 2014(1), p.692076.

Khalid, S., Khalil, T. and Nasreen, S., (2014), August. A survey of feature selection and feature extraction techniques in machine learning. In 2014 science and information conference (pp. 372-378). IEEE.

Khan, M.A., (2021). HCRNNIDS: Hybrid convolutional recurrent neural network-based network intrusion detection system. Processes, 9(5), p.834.

Khraisat, A., Gondal, I., Vamplew, P. and Kamruzzaman, J., (2019). Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity, 2(1), pp.1-22.

Kim, J., Kim, J., Thu, H.L.T. and Kim, H., (2016), February. Long short term memory recurrent neural network classifier for intrusion detection. In 2016 international conference on platform technology and service (PlatCon) (pp. 1-5). IEEE.

Kim, T.K., (2017). Understanding one-way ANOVA using conceptual figures. Korean journal of anesthesiology, 70(1), pp.22-26.

Kimmel, J.C., Mcdole, A.D., Abdelsalam, M., Gupta, M. and Sandhu, R., (2021). Recurrent neural networks based online behavioural malware detection techniques for cloud infrastructure. IEEE Access, 9, pp.68066-68080.

Koay, A., Chen, A., Welch, I. and Seah, W.K., (2018), January. A new multi classifier system using entropy-based features in DDoS attack detection. In 2018 International conference on information networking (ICOIN) (pp. 162-167). IEEE.

Kousar, H., Mulla, M.M., Shettar, P. and Narayan, D.G., (2021), June. Detection of DDoS attacks in software defined network using decision tree. In 2021 10th IEEE international conference on Communication Systems and Network Technologies (CSNT) (pp. 783-788). IEEE.

Kshirsagar, P.R., Yadav, R.K. and Patil, N.N., (2022). Intrusion detection system attack detection and classification model with feed-forward lstm gate in conventional dataset. Machine Learning Applications in Engineering Education and Management, 2(1), pp.20-29.

Kumar, D., Pateriya, R.K., Gupta, R.K., Dehalwar, V. and Sharma, A., (2023). DDoS detection using deep learning. Procedia Computer Science, 218, pp.2420-2429.

Larijani, H., Ahmad, J. and Mtetwa, N., (2018), September. A novel random neural network based approach for intrusion detection systems. In 2018 10th Computer Science and Electronic Engineering (CEEC) (pp. 50-55). IEEE.

Li, Q., Huang, H., Li, R., Lv, J., Yuan, Z., Ma, L., Han, Y. and Jiang, Y., (2023). A comprehensive survey on DDoS defense systems: New trends and challenges. Computer Networks, p.109895.

Li, W., Qi, F., Tang, M. and Yu, Z., (2020). Bidirectional LSTM with self-attention mechanism and multi-channel features for sentiment classification. Neurocomputing, 387, pp.63-77.

Li, Y. and Liu, Q., (2021). A comprehensive review study of cyber-attacks and cyber security; Emerging trends and recent developments. Energy Reports, 7, pp.8176-8186.

Lin, S.W., Ying, K.C., Lee, C.Y. and Lee, Z.J., (2012). An intelligent algorithm with feature selection and decision rules applied to anomaly intrusion detection. Applied Soft Computing, 12(10), pp.3285-3290.

Liu, M., Xue, Z., Xu, X., Zhong, C. and Chen, J., (2018). Host-based intrusion detection system with system calls: Review and future trends. ACM computing surveys (CSUR), 51(5), pp.1-36.

Loukas, G., Vuong, T., Heartfield, R., Sakellari, G., Yoon, Y. and Gan, D., (2017). Cloud-based cyber-physical intrusion detection for vehicles using deep learning. IEEE Access, 6, pp.3491-3508.

Maithem, M. and Al-Sultany, G.A., (2021), February. Network intrusion detection system using deep neural networks. In Journal of Physics: Conference Series (Vol. 1804, No. 1, p. 012138). IOP Publishing.

Malik, M. and Dutta, M., (2023). Feature engineering and machine learning framework for DDoS attack detection in the standardized internet of things. IEEE Internet of Things Journal, 10(10), pp.8658-8669.

Mazini, M., Shirazi, B. and Mahdavi, I., (2019). Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms. Journal of King Saud University-Computer and Information Sciences, 31(4), pp.541-553.

Meftah, S., Rachidi, T. and Assem, N., (2019). Network based intrusion detection using the UNSW-NB15 dataset. International Journal of Computing and Digital Systems, 8(5), pp.478-487.

Meng, Y., (2012), July. Measuring intelligent false alarm reduction using an ROC curve-based approach in network intrusion detection. In 2012 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications (CIMSA) Proceedings (pp. 108-113). IEEE.

Mirjalili, S., (2016). Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural computing and applications, 27, pp.1053-1073.

Moradkhani, M., Amiri, A., Javaherian, M. and Safari, H., (2015). A hybrid algorithm for feature subset selection in high-dimensional datasets using FICA and IWSSr algorithm. Applied Soft Computing, 35, pp.123-135.

Najafimehr, M., Zarifzadeh, S. and Mostafavi, S., (2023). DDoS attacks and machine-learning- based detection methods: A survey and taxonomy. Engineering Reports, 5(12), p.e12697.

Navruzov, E. and Kabulov, A., (2022), June. Detection and analysis types of DDoS attack. In 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS) (pp. 1-7). IEEE.

Niu, Z., Zhong, G. and Yu, H., (2021). A review on the attention mechanism of deep learning. Neurocomputing, 452, pp.48-62.

Nuiaa, R.R., Manickam, S. and Alsaeedi, A.H., (2021). Distributed reflection denial of service attack: A critical review. International Journal of Electrical and Computer Engineering, 11(6), p.5327.

Osanaiye, O., Cai, H., Choo, K.K.R., Dehghantanha, A., Xu, Z. and Dlodlo, M., (2016). Ensemble-based multi-filter feature selection method for DDoS detection in cloud computing. EURASIP Journal on Wireless Communications and Networking, 2016, pp.1-10.

Pacheco, J., Benitez, V. and Félix, L., (2019), July. Anomaly behavior analysis for IoT network nodes. In Proceedings of the 3rd International Conference on Future Networks and Distributed Systems (pp. 1-6).

Patani, N. and Patel, R., (2017). A mechanism for prevention of flooding based ddos attack. International Journal of Computational Intelligence Research, 13(1), pp.101-111.

Pham, N.T., Foo, E., Suriadi, S., Jeffrey, H. and Lahza, H.F.M., (2018), January. Improving performance of intrusion detection system using ensemble methods and feature selection. In Proceedings of the Australasian computer science week multiconference (pp. 1-6).

Qian, X., Zhang, C., Chen, L. and Li, K., (2022). Deep learning-based identification of maize leaf diseases is improved by an attention mechanism: Self-attention. Frontiers in Plant Science, 13, p.864486.

Rahman, C.M., Rashid, T.A., Alsadoon, A., Bacanin, N., Fattah, P. and Mirjalili, S., (2023). A survey on dragonfly algorithm and its applications in engineering. Evolutionary Intelligence, 16(1), pp.1-21.

Raj, J.S., (2019). A comprehensive survey on the computational intelligence techniques and its applications. Journal of ISMAC, 1(03), pp.147-159.

Rajasekharaiah, K.M., Dule, C.S. and Sudarshan, E., (2020), December. Cyber security challenges and its emerging trends on latest technologies. In IOP Conference Series: Materials Science and Engineering (Vol. 981, No. 2, p. 022062). IOP Publishing.

Ravi, V., Chaganti, R. and Alazab, M., (2022). Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system. Computers and Electrical Engineering, 102, p.108156.

Ravindranath, V., Ramasamy, S., Somula, R., Sahoo, K.S. and Gandomi, A.H., (2020), July. Swarm intelligence based feature selection for intrusion and detection system in cloud infrastructure. In 2020 IEEE Congress on Evolutionary Computation (CEC) (pp. 1-6). IEEE..

Raza, M.S., Sheikh, M.N.A., Hwang, I.S. and Ab-Rahman, M.S., (2024), April. Feature-Selection-Based DDoS Attack Detection Using AI Algorithms. In Telecom (Vol. 5, No. 2, pp. 333-346). MDPI.

Rezaeipanah, A., Mousavipoor, S.E., Asayeshjoo, M. and Sadeghzadeh, M., (2021). Combining Particle Swarm Optimization and Entropy to Detect DDoS Attacks in the Cloud Computing. Journal of Business Data Science Research, 1(01), pp.33-43.

Ring, M., Wunderlich, S., Scheuring, D., Landes, D. and Hotho, A., (2019). A survey of network-based intrusion detection data sets. Computers & security, 86, pp.147-167.

Rosay, A., Carlier, F. and Leroux, P., (2020), May. Feed-forward neural network for Network Intrusion Detection. In 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring) (pp. 1-6). IEEE.

Rouder, J.N., Engelhardt, C.R., McCabe, S. and Morey, R.D., (2016). Model comparison in ANOVA. Psychonomic bulletin & review, 23, pp.1779-1786.

Rudro, R. A. M., SOHAN, M. F. A. A., Chaity, S. K., and Reya, R. I, (2023). Enhancing DDoS Attack Detection Using Machine Learning: A Framework with Feature Selection and Comparative Analysis of Algorithms. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 4(03), 1185–1192

Saini, P.S., Behal, S. and Bhatia, S., (2020), March. Detection of DDoS attacks using machine learning algorithms. In 2020 7th International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 16-21). IEEE.

Salman Iqbal, Miss Laiha Mat Kiah, Babak Dhaghighi, Muzammil Hussain, Suleman Khan, Muhammad Khurram Khan, Kim-Kwang Raymond Choo, (2016), On cloud security attacks: A taxonomy and intrusion detection and prevention as a service, Journal of Network and Computer Applications, 74, pp.98-120.

Saranya, T., Sridevi, S., Deisy, C., Chung, T.D. and Khan, M.A., (2020). Performance analysis of machine learning algorithms in intrusion detection system: A review. Procedia Computer Science, 171, pp.1251-1260.

Satilmiş, H., Akleylek, S. and Tok, Z.Y., (2024). A Systematic Literature Review on Host-Based Intrusion Detection Systems. Ieee Access, 12, pp.27237-27266.

Sayed, G.I., Tharwat, A. and Hassanien, A.E., (2019). Chaotic dragonfly algorithm: an improved metaheuristic algorithm for feature selection. Applied Intelligence, 49, pp.188-205.

Seth, J.K. and Chandra, S., (2018). An effective DOS attack detection model in cloud using artificial bee colony optimization. 3D Research, 9, pp.1-13.

Shamshirband, S., Fathi, M., Chronopoulos, A.T., Montieri, A., Palumbo, F. and Pescapè, A., (2020). Computational intelligence intrusion detection techniques in mobile cloud computing environments: Review, taxonomy, and open research issues. Journal of Information Security and Applications, 55, p.102582.

Sharafaldin, I., Lashkari, A.H., Hakak, S. and Ghorbani, A.A., (2019), October. Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy. In 2019 international carnahan conference on security technology (ICCST) (pp. 1-8). IEEE.

Sharifian, Z., Barekatain, B., Quintana, A.A., Beheshti, Z. and Safi-Esfahani, F., (2023). Sin-Cos-bIAVOA: A new feature selection method based on improved African vulture optimization algorithm and a novel transfer function to DDoS attack detection. Expert Systems with Applications, 228, p.120404.

Shen, Y., Zheng, K., Wu, C., Zhang, M., Niu, X. and Yang, Y., (2018). An ensemble method based on selection using bat algorithm for intrusion detection. The Computer Journal, 61(4), pp.526-538.

Singh, C. and Jain, A.K., (2024). A Comprehensive Survey on DDoS Attacks Detection & Mitigation in SDN-IoT Network. e-Prime-Advances in Electrical Engineering, Electronics and Energy, p.100543.

Singh, G. and Khare, N., (2022). A survey of intrusion detection from the perspective of intrusion datasets and machine learning techniques. International Journal of Computers and Applications, 44(7), pp.659-669.

Staudemeyer, R.C., (2015). Applying long short-term memory recurrent neural networks to intrusion detection. South African Computer Journal, 56(1), pp.136-154.

Tandon, R., (2020). A survey of distributed denial of service attacks and defenses. arXiv preprint arXiv:2008.01345.

Tesfahun, A. and Bhaskari, D.L., (2013), November. Intrusion detection using random forests classifier with SMOTE and feature reduction. In 2013 International conference on cloud & ubiquitous computing & emerging technologies (pp. 127-132). IEEE.

Venkatesh, B. and Anuradha, J., (2019). A review of feature selection and its methods. Cybernetics and information technologies, 19(1), pp.3-26.

Vinayakumar, R., Alazab, M., Soman, K.P., Poornachandran, P., Al-Nemrat, A. and Venkatraman, S., (2019). Deep learning approach for intelligent intrusion detection system. Ieee Access, 7, pp.41525-41550.

Vinayakumar, R., Soman, K.P. and Poornachandran, P., (2017), September. Evaluating effectiveness of shallow and deep networks to intrusion detection system. In 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (pp. 1282-1289). IEEE.

Wang, A., Chang, W., Chen, S. and Mohaisen, A., (2018). Delving into internet DDoS attacks by botnets: characterization and analysis. IEEE/ACM Transactions on Networking, 26(6), pp.2843-2855.

Wang, H., Wu, J., Zhang, C., Lu, W. and Ni, C., (2024). Intelligent Security Detection and Defense in Operating Systems Based on Deep Learning. International Journal of Computer Science and Information Technology, 2(1), pp.359-367.

Wu, S.X. and Banzhaf, W., (2010). The use of computational intelligence in intrusion detection systems: A review. Applied soft computing, 10(1), pp.1-35.

Xu, C., Shen, J., Du, X. and Zhang, F., (2018). An intrusion detection system using a deep neural network with gated recurrent units. IEEE Access, 6, pp.48697-48707.

Yazdani, M. and Jolai, F., (2016). Lion optimization algorithm (LOA): a nature-inspired metaheuristic algorithm. Journal of computational design and engineering, 3(1), pp.24-36.

Yu, L. and Liu, H., (2003). Feature selection for high-dimensional data: A fast correlation-based filter solution. In Proceedings of the 20th international conference on machine learning (ICML-03) (pp. 856-863).

Yu, X., Han, D., Du, Z., Tian, Q. and Yin, G., (2019). Design of DDoS attack detection system based on intelligent bee colony algorithm. International Journal of Computational Science and Engineering, 19(2), pp.223-232.

Yu, Z., Gao, H., Cong, X., Wu, N. and Song, H.H., (2023). A Survey on Cyber–Physical Systems Security. IEEE Internet of Things Journal, 10(24), pp.21670-21686.

Zhang, H., Li, Y., Lv, Z., Sangaiah, A.K. and Huang, T., (2020). A real-time and ubiquitous network attack detection based on deep belief network and support vector machine. IEEE/CAA Journal of Automatica Sinica, 7(3), pp.790-799.

Zhao, Z., Li, Z., Zhou, Z., Yu, J., Song, Z., Xie, X., Zhang, F. and Zhang, R., (2024). DDoS family: A novel perspective for massive types of DDoS attacks. Computers & Security, 138, p.103663.

Zhong, M., Lin, M., Zhang, C. and Xu, Z., (2024). A Survey on Graph Neural Networks for Intrusion Detection Systems: Methods, Trends and Challenges. Computers & Security, p.103821.

Zhou, L., Zhu, Y., Xiang, Y. and Zong, T., (2023). A novel feature-based framework enabling multi-type DDoS attacks detection. World Wide Web, 26(1), pp.163-185.

Zulhilmi, A., Mostafa, S.A., Khalaf, B.A., Mustapha, A. and Tenah, S.S., (2021). A comparison of three machine learning algorithms in the classification of network intrusion. In Advances in Cyber Security: Second International Conference, ACeS 2020, Penang, Malaysia, December 8-9, 2020, Revised Selected Papers 2 (pp. 313-324). Springer Singapore.

Website References:

1. 2024 Global Threat Analysis Report, https://www.radware.com/threat-analysis-report/
2. CAIDA Available at https://www.caida.org/catalog/datasets/completed-datasets/
3. CICDDoS2019 Available at https://www.unb.ca/cic/datasets/ddos-2019.html
4. CICIDS2017 Available at https://www.unb.ca/cic/datasets/ids-2017.html
5. CIDD Available at http://groups.di.unipi.it/~hkholidy/projects/cidd/
6. CIDDS-001 Available at https://www.hs-coburg.de/forschung/forschungsprojekte-oeffentlich/informationstechnologie/cidds-coburg-intrusion-detection-data-sets.html
7. CSE-CIC-IDS2018 Available at https://www.unb.ca/cic/datasets/ids-2018.html
8. DARPA KDD CUP 1999, Available at https://www.kdd.org/kdd-cup/view/kdd-cup-1999/Tasks
9. Data mining software in Java. http://www.cs.waikato.ac.nz/ml/weka/
10. ISOT-CID Available at https://onlineacademiccommunity.uvic.ca/isot/datasets/
11. NSL-KDD Available at https://www.unb.ca/cic/datasets/nsl.html
12. UNSW-NB15 Available at https://research.unsw.edu.au/projects/unsw-nb15-dataset
13. Uspscims Available at https://github.com/uspscims/cloudsecurity
14. A Retrospective on DDoS Trends in 2023 and Actionable Strategies for 2024, https://www.akamai.com/blog/security/a-retrospective-on-ddos-trends-in-2023

Case Studies References:

Amazon Web Services, (2023). AWS Shield Advanced: DDoS protection. Amazon Web Services Documentation. Available at: https://docs.aws.amazon.com/waf/latest/developerguide/ddos-overview.html

Lashkari, A.H., Draper-Gil, G., Mamun, M. and Ghorbani, A.A., (2017). CICFlowMeter: Network traffic flow generator for intrusion detection systems. University of New Brunswick, Canadian Institute for Cybersecurity. Available at: https://www.unb.ca/cic/research/applications.html

Google Cloud, (2023). Adaptive Protection with Cloud Armor. Google Cloud Documentation. Available at: https://cloud.google.com/armor/docs/adaptive-protection-overview

PUBLICATIONS

Journals

i. Kalaivani, M., Padmavathi, G. 'Panthera Leo Optimized Multilayer Feed Forward Learning-Based Intrusion Detection Model for Cloud." SN COMPUT. SCI. 4, 800 (2023). https://doi.org/10.1007/s42979-023-02225-x, Springer Nature

ii. Kalaivani M and Padmavathi G, "Ensembling of Attention-based Recurrent Units for Detection and Mitigation of Multiple Attacks in Cloud" International Journal of Advanced Computer Science and Applications(IJACSA), 14(10), 2023. http://dx.doi.org/10.14569/IJACSA.2023.0141010 Indexed in Scopus

Conferences

i. Kalaivani M, Padmavathi G, "Strategic-level framework with combined feature engineering and optimization Methods for detecting DDoS Flooding attacks in Cloud Environment", CEMC 2022, October 29-30, 2022, published in the Grenze International Journal of Engineering and Technology (GIJET) as Volume 9 Issue 1 (January 2023 issue), https://thegrenze.com/index.php?display=page&view=journaldetails&id=8 . Indexed in Scopus.

ii. Kalaivani M, Padmavathi G, "Hybrid Ensemble based feature engineering techniques for detecting direct DDOS flooding attack in Cloud", MLIP-2022, November 2022, published in the Grenze International Journal of Engineering and Technology (GIJET) as Volume 9 Issue 1 (January 2023 issue), https://thegrenze.com/index.php?display=page&view=journaldetails&id=8 . Indexed in Scopus

Avinashilingam Institute for HomeScience and Higher Education for Women
(Deemed to be University Estd. u/s 3 of UGC Act 1956, Category 'A' by MHRD
Re-accredited with A++ Grade by NAAC. CGPA 3.65/4, Category I by UGC
Coimbatore - 641 043, Tamil Nadu, India

Appendix L2

(Item No 5 of Check List) Details of Research Publications

S.No	Article	Journal	Other Details Vol/No/Page No/ Year	Published in UGC-CARE / Scopus Indexed/ Web of Science
1	Pantnera Leo optimized Multilayer Feed Forward Learning-Based Intrusion detection Model for cloud.	SN Computer Science	4(6), 800. 2023	Scopus Indexed
2	Ensembling of Attention-based Recurrent Units for Detection and mitigation of multiple attacks in cloud	International Journal of Advanced Computer Science and Applications	14(10), 86-92. 2023	Scopus Indexed

*Proof of list of Journals from Internet to be attached along with copies of reprints.

Scholar : dhuny.

Supervisor : M. Mahendran 29/05/2024

The Scholar Miss. Kalaivani, M (18PHCSP005) has published her article in the following journals:

Checked By: M. Mahendran 29/05/2024
HoD/Dean of Respective School

1. SN Computer Science - indexed and active in Scopus from 2020 to

present and

2. International Journal of Advanced Computer Science and Applications- indexed and active in scopus from 2017 to present.

This may be considered.

29.05.24

SN Computer Science (2023) 4:800
https://doi.org/10.1007/s42979-023-02225-x

ORIGINAL RESEARCH

Panthera Leo Optimized Multilayer Feed Forward Learning-Based Intrusion Detection Model for Cloud

M. Kalaivani[1] · **G. Padmavathi**[1]

Received: 24 July 2023 / Accepted: 5 August 2023

Abstract

Security threats are growing at an exponential speed paralyzing the privacy measures among the users. Because of this, maintaining client security protection remains to be a challenging task for analysts. Recently, machine learning (ML) algorithms has gained the major light of importance while creating a successful intrusion detection system (IDS) to ensure the security in the network. However, these machine learning algorithms-based IDS suffers from the high chance of non-detecting failures due to the large complex and variable data sets. To defeat these previously mentioned issues, this paper proposes the novel single feedforward network ensembled with Panthera Leo Optimization (PLO) to ensure the high performance in detecting the different vulnerabilities in Cloud network. The CIDCC data sets-2019 are used to conduct extensive testing, and a number of performance metrics, including accuracy, precision, recall, specificity, and F1-score, are computed. The proposed IDS is contrasted and the other state-specialty of learning model-based IDS, for example, Support Vector Machines (SVM), Ensembled machine learning algorithm (EML), Classification and Regression Trees (CART), and Bat Optimized Extreme Learning Machines (BEL) calculation. According to simulation results, the suggested has done better than the alternate learning model with regard to high accuracy (98.65%), accuracy (98.62%), recall (98.65%), specificity (98.62%), and F1-Score (98.6%), which shows higher vulnerability to expanding network threats.

Keywords Cloud security · CIDCC data sets · Panthera Leo optimization · Single feed forward layers

Introduction

Today, a profusion of Innovative and integrated technologies is transforming our pleasant lives. In this sense, cloud is quickly establishing itself as a cutting-edge technology that may alter daily life into one that is more intelligent [1–4]. The predicted interconnection of 45.8 billion cloud setups by 2025 presents several challenges [5]. Setting up a cloud storage is thought to provide the most issues in terms of security and privacy preservation.

Traditional security measures include static perimeter network defences (such as firewalls and IDS) and widespread end-user deployment of antivirus software. However, because of the variety in device features, these processes are unable to create scalable cloud networks [6–10]. In addition, as the cloud network expands, so does the number of attacks, which makes it impossible to now comply with [11]. Many frequent interruption location frameworks, such as SNORT and BRO [12], are limited to IP packets, since their components are static and signature-based. Other metaheuristic algorithms exhibit high exploration and exploitation [13, 14]. However, the absence of intelligence in these approaches allows assaults to ruin the network's whole structure. Therefore, the intrusion detection system's mechanism has to be improved so as to get a greater detection mechanism that can counteract attacks in Cloud networks.

Using this information as a foundation, we created a unique IDS architecture based on Ensembled Panthera Leo Optimization (PLO) Feedforward learning machines for

This article is part of the topical collection "Advances in Computational Approaches for Image Processing, Wireless Networks, Cloud Applications and Network Security" guest edited by P. Raviraj, Maode Ma and Roopashree H R.

✉ M. Kalaivani
kalaivanim@gmail.com

G. Padmavathi
padmavathi.avinashilingam@gmail.com

[1] Department of Computer Science, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, India

Published online: 14 October 2023

enhanced attack prediction from hostile nodes. The study also demonstrates that several hazardous conditions were used to assess the applicability of the suggested methodology. The implementation of the Ensembled PLO-Learning algorithm-based IDS in Cloud network is discussed in this research. This method has been shown to be reliable and to maintain steady detection performance even when the quantity of hostile nodes increases.

According to our knowledge, the suggested IDS with Optimized Extreme Learning machines here has a revolutionary and unique architecture that addresses the majority of the aforementioned weaknesses of the existing systems. This paper's contribution is as follows:

1. The implementation of an IDS based on High Efficient Ensembled Feed-Forward Machine Learning that can handle bigger attack data sets.
2. The CIDCC Data sets-2019, which consist of 41 characteristics and several attack types, are used to validate the proposed technique.
3. The experiment is run utilizing the aforementioned data sets, where different performance measures are computed and examined.

This paper is organised as, with "Related Works " discussing several authors' relevant efforts on intrusion detection systems that rely on machine learning. "System Overview" discusses the suggested framework, implementation model, data set creation, and classifiers' recommended process. In "Results with its Discussion", results with comparative analysis were provided. Future scope and the conclusion are covered in "Conclusion Scope".

Related Works

The use of supervised learning approaches for locating network stabbings in social networks is explained in this section.

The author recommends a rule's development technique built on Genetic Programming for identifying fresh network threats [15]. "Reproduction, mutation, crossover, & dropping condition operators" are four genetic operator, utilized in their methodology to generate new rules. To automatically recognize or detect network hazards, new rules are used. It has been demonstrated through experiments that rules developed by GPs using data from the KDD 1999 Cup had a high rate of recognizing unidentified assaults while having low false positive and false negative rates.

A hybrid approach which includes the SVM, decision tree, and NB algorithms was developed by Goeschel et al. [16]. They divided the data set into typical and abnormal cases using an SVM model. They identified particular attack types in the anomalous samples using a decision tree technique. On the other hand, a classification tree technique can only identify known assaults and not undiscovered ones. They thus employed a Nave Bayes classifier to uncover previously unknown assaults.

Kuttranont et al. [17] suggested a K-nearest neighbor identification approach that runs on a GPU employing massively parallel techniques for a quicker computation. They altered the KNN algorithm's neighbor selecting rule. The optimization approach selects a current proportion (such as 50%) of the surrounding samples rather than first K closest samples, which are selected as neighbors by the standard KNN. The present method works well with sparse data and takes into account abnormalities in data distribution.

A GAN was utilized by Zhang et al. [18] to complement their data. Due to the imbalance and lack of fresh data in the "KDD99" data set, machine learning models produced as a consequence have poor generalizability. They used a GAN to add extra data to the collection to solve these issues. The flow data from KDD99 was duplicated in the GAN model's data. Attack variants can be discovered by incorporating this acquired data into the training set. To compare the accuracy of the two data sets, eight different types of assaults were used. According to the trials, adversarial learning increased 7 accuracy in 8 attack types.

Weak learners are classifiers that just marginally outperform a random classifier. Many weak learners are merged to create ensemble classifiers, which greatly enhance the performance of a classifier. A few common techniques for merging poor learners are majority voting, bagging, and boosting [19]. Even though the ensemble classifier combines the drawbacks of component classifiers, it has in some combinations achieved extremely effective results. Lightweight intrusion detection using supervised learning was proposed by Jan et al. [20]. SVM classifier was created by them to identify assaults (target DDoS). Anomaly and attack detection were studied by Hasan et al. [21]. They utilized ML methods including LR, SVM, decision tree, RF & ANN to carry out their research.

System Overview

The proposed architecture consists of three tiers of working. In the first tier, different data sets under the normal and malicious environment are collected. The collected data sets are pre-processed which suits for the training network. The features were extracted in the third tier and finally the proposed framework is implemented for an effective prediction of normal and malicious nodes. As a result, one of the features in the event of an attack is the system's output found in the data sets that have been gathered: (1) the device address

that is being attacked; (2) whether the device is malicious; and (3) the sort of attack that has taken place.

Data Collection

The Canadian Institute for Cybersecurity (CIC) developed the data set utilized in the proposed network in 2017. Common attacks are included in the CIC IDS data set (CIC-IDS2017), representative of real-world data. Generated Labelled Flows and Machine Learning CVE are the two files that make up this data set; the former has 86 features, while the latter has 79. This study uses a Machine Learning CSV data file with eight traffic monitoring sessions spread over 5 days. For further analysis, 1 CSV file is created by combining these 8 files. The merged file has 2,830,743 rows, 78 features columns, with 1 label column.

Data Pre-processing

Before deploying in the proposed model, data pre-processing is necessary. From the data sets, it is observed that the 2867 rows consist of infinity values and NAN which should be removed. The resultant data sets consists of 2827 rows of valid data which are then used for the training the proposed framework.

Feature Extraction

With larger data sets, overfitting is the prime problem which affects the network's performance of prediction. The regularization is common method which is used to overcome the above problem. For an efficient feature extraction method, this research work uses ANOVA (analysis of variance) and Chi-squared correlation techniques. Due to its high-speed characteristics on the data filtering, above methods are used in this research for extraction of top features from the pre-processed data. This method extracted 15 mandatory features from the whole data sets which reflects the whole characteristics of the data sets.

Proposed Model

The Single Feedforward Layer Network (SLFN) optimal with Panthera Leo Optimization techniques is used in the suggested framework. The SLFN and Panthera Leo Optimization, which are used to categorize different attacks, have the Extreme Learning Machines idea incorporated. Below is a discussion of the two algorithms' preliminary overviews.

Single Feed-Forward Layers

The research's implementation of a single feedforward layer is based on ELM. ELM were proposed by Huang [22], and they use a single hidden layer, a network with high preparation and execution speeds, outstanding speculative/exactness, and capabilities for approximating any function [22].

In this type of system, the target layer's activation function is straight, while the "L" neurons in the concealed layer operating with an activation function that is remarkably differential (for example, the sigmoid function). Concealed layers do not need to be tuned manually in ELM. ELM does not need that the hidden layer is tweaked.

Counting the biàs loads, the hidden layer's loads are chosen arbitrary. Although it is not accurate to say that concealed nodes are useless, hidden neurons' parameters can be produced at random, even beforehand.

Before handling the training set data, that is.

The system yield is given by equation for an ELM with a single concealed layer:

$$f_L(x) = \sum_{i=1}^{L} \beta_i h_i(x) = h(x)\beta, \tag{1}$$

where $x \rightarrow$ input, $\beta \rightarrow$ The target weight vector is provided as follows:

$$\beta = [\beta_1, \beta_2, \ldots\ldots\ldots\ldots\ldots \beta_L]^T. \tag{2}$$

$H(x) \rightarrow$ output concealed layer is given by

$$h(x) = [h_1(x), h_2(x), \ldots\ldots\ldots\ldots\ldots h_L(x). \tag{3}$$

The hidden layers are represented by Eq. (4), which is used to determine output vector O (target vector):

$$H = \begin{bmatrix} h(x_1) \\ h(x_2) \\ \vdots \\ h(x_N) \end{bmatrix}. \tag{4}$$

The basic implementation of the ELM makes use of nonlinear least squares techniques, as shown in the following equation:

$$\beta' = H^* O = H^T (HH^T)^{-1} O, \tag{5}$$

where $H*$ Moore−Penrose generalized inverse (inverse of H).

The equation is provided as

$$\beta' = H^T \left(\frac{1}{C} HH^T \right)^{-1} O. \tag{6}$$

Consequently, the output function may be determined using the equation above:

$$f_L(x) = h(x)\beta = h(x)H^T\left(\frac{1}{C}HH^T\right)^{-1}O. \tag{7}$$

ELM incorporates the ability of the piece to produce outstanding precision for better presentation. Less preparation error and better estimation are the main advantages of the ELM. ELM has applications in forecast values, because it makes use of the manual tuning of weight predispositions and non-zero actuation capabilities [22] gives a more realistic representation of ELM's circumstances. In Algorithm 1, the ELM's pseudo-code is shown.

In step one Training sets of "N" data with an activation function and "*n*" hidden neurons.

In step two, biases and input weights are assigned.

Calculate the hidden matrix H in step three.

Calculate the output weight matrix in step four.

Classify or predict the values in Step 5.

Motivation

Although ELM proves to be useful for both testing and training, a major barrier is the non-ideal tuning of information loads and predispositions. When compared to other conventional learning methods, ELM also uses a number of hidden layers to adjust the ideal loads, which may affect the forecast's accuracy. Another Panthera Leo is utilized to streamline the hyper parameters, such as input loads and predisposition factors, to create the high precision of the order to overcome the aforementioned disadvantage.

Panthera Leo Optimizer (PLO)

The hunting nature of lions inspires Panthera Leo's Optimization. Lions usually live in groups called Pride (Q). Lionesses usually hunt in groups, with many lionesses working together to encircle the prey from various angles to trap it. Male lions and a few lionesses rest and await the arrival of hunter lionesses. A fixed number of females in each Q seek prey in a group to feed the Q members. As shown in Fig. 1, these hunter lions use specific techniques to encircle and catch their prey. During hunting, lions generally follow a similar routine. Every lioness modifies her location while hunting based on its own or other members' positions. As a result, some hunting lions surround and assault the victim from the other direction, and LOA employs opposition-based learning (OBL). The seekers are divided into three "wings," with the central wing having the highest cumulative fitness and the left, and right wings fixed randomly.

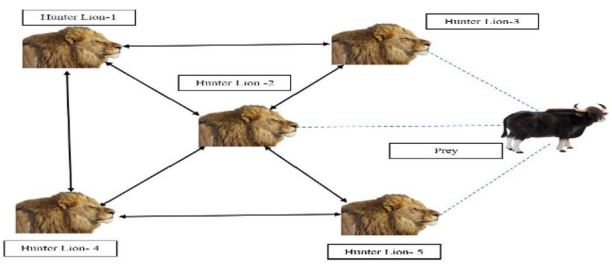


Fig. 1 Encircling of prey as per Panthera Leo's Pride (Q)

When a hunter improves his or her fitness, the prey flees to a new place, as depicted in the following equation:

$$QZ' = QZ + \text{rand}(0,1) \times QZI \times (QZ - \text{Hunter}), \tag{8}$$

QZ represents the prey's current location, Hunter represents a new hunter position who attacks the prey, and QZI represents the percent increase in the hunter's fitness. The following is the new location of the hunters belonging to the left and right wings:

$$\begin{aligned}\text{Hunter}' = \{&\text{rand}((2 \times QZ - \text{Hunter}), QZ), (2 \times QZ - \text{Hunter}) \\ &< QZ\text{rand}((2 \times QZ - \text{Hunter})), (2 \times QZ - \text{Hunter}) > QZ.\end{aligned} \tag{9}$$

Global hunters' selection of new positions was estimated using Eq. (9):

$$\begin{aligned}\text{Hunter}' = \{&\text{rand}(\text{Hunter}, QZ), \text{Hunter} < QZ\text{rand}(QZ, \text{Hunter}), \\ &\text{Hunter} > QZ.\end{aligned} \tag{10}$$

Among Hunters and prey, Rand (Hunter, QZ) provides a random number. This hunting attitude has several advantages in terms of obtaining better solutions. This technique creates a circle-shaped community around the prey, causing hunters to approach the animal from all sides. This hunting procedure of the Panthera Leo is used to tune the hyperparameters of ELM.

Merits of PLO Algorithms

1. Compared to other meta-heuristic algorithms, such as the Grey wolf optimizer, genetic algorithms, and even particle swarm optimization, it exhibits high exploration and exploitation.
2. Less time complexity.

Table 1 Arithmetic equation for the performance standards calculation

Sl.no.	Performance standards	Equations
01	Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
02	Sensitivity or recall	$\frac{TP}{TP+FN} \times 100$
03	Specificity	$\frac{TN}{TN+FP}$
04	Precision	$\frac{TN}{TP+FP}$
05	F1-Score	$\frac{Precison*Recall}{Precision+Recall}$

"False Positive, True Negative, False Positive, and False Negative values" are abbreviated as TP, TN, FP, and FN, respectively

Proposed Model

Training ELM cells is done to determine the ideal loads and predisposition esteem, as was discussed above. Two steps must be accomplished to form the problem of preparing the ELM cells using Panthera Leo improvement.

- A good strategy for dealing with the propensity burdens in ELM cells.
- Create the fitness function.

The main purpose of developing the feedforward, learning model is to reach the most notable agreement regarding expectation/identification exactness for both preparation and testing tests. The fixing of goal capacity is its most important component. The suggested model stipulates that the ELM network's wellness capability is provided by

$$\text{fitness function} = \text{Maximum (Accuracy)}. \quad (11)$$

Typically, mean square error, whose outcomes are decided by numerical articulation, is used to estimate the presentation of learning models (18). The wellness capacity is set to the highest level of expectation accuracy. Table V displays the information that was utilised to construct the suggested model. Algorithm-1 introduces the proposed PLO-overall ELM model's operation.

S no	Algorithm1 // Pseudo Code for the Proposed PLO-ELM-IDS model
01	Input = {f1, f2, f3, f4,.....,f10) Assume f as input attributes
02	Output: Identification of the attacks
03	Epochs present: 100
04	the weights and bias for the input as taken at random
05	Start the while loop
06	Measure output from the ELM network by utilizing equation (7)
07	Measure FF with equation (11)
08	Check for (FF greater than threshold)
09	Start the For loop till max. iteration
10	the bias weights & input layers are assigned by equation (8) & (9)
11	Measure the output FF using equation (11)
12	Check for (FF equal to threshold)
13	Jump Step 16
14	otherwise
15	Jump Step 9
16	halt
17	halt

Fig. 2 ROC curve for the proposed algorithm using CIDCC data sets—2017 (70:30)

Fig. 3 ROC curve for the proposed algorithm using CIDCC data sets—2017 (80:20)

Label	*Normal*	*Malicious*
Normal	98.64%	1.2%
Malicious	1.2%	98.67%

(a)

Label	Normal	Malicious
Normal	98.64%	1.2%
Malicious	1.2%	98.60%

(b)

Fig. 4 Confusion matrix for the proposed algorithm (**a**) 70:30 data sets (**b**) 80:20 data sets

Results with Its Discussion

The entire exploratory setup was run on a workstation with an i9 CPU, 16 GB of RAM, and Nvidia Titan Board, 256 GB. The Keras API and TensorFlow 2.1 were used to develop the proposed deep—learning calculation.

Performance Standards

The performance standards accuracy, precision, recall, specificity and F1-score are used to assess the proposed model and are calculated using the mathematical expressions presented in Table 1.

Results

This section introduces the advantages of the suggested design over the other models already in use. Tri-folds of

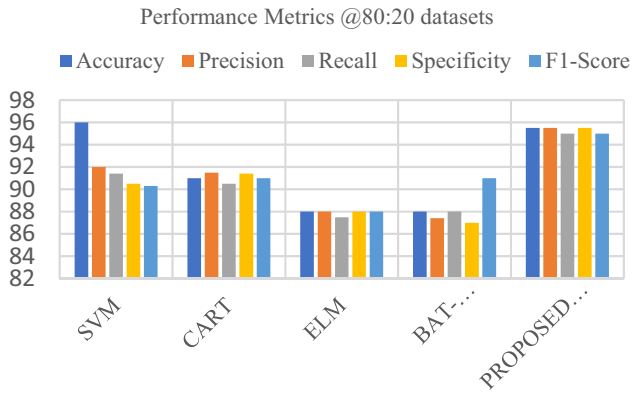


Fig. 5 Comparative analysis of various IDS at the ratio of 80:20 data sets

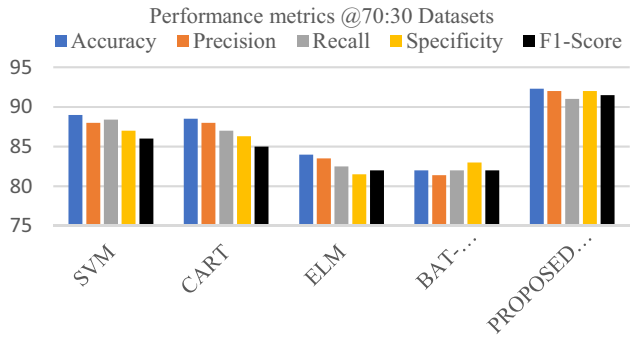


Fig. 6 Comparative analysis of the different IDS at the ratio of 70:30 data sets

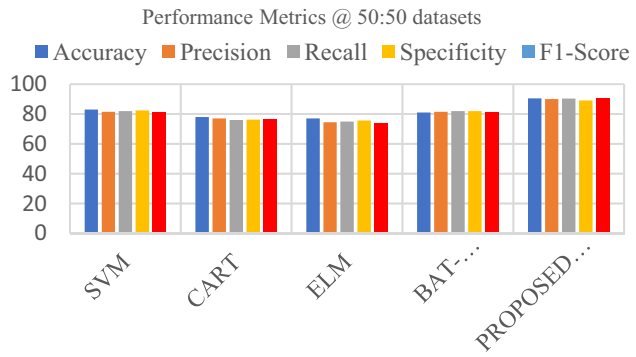


Fig. 7 Comparative analysis of various IDS at the ratio of 50:50 data sets

the suggested design are validated. Disarray lattice is used in the main crease to support the presentation of the suggested engineering. In addition, Receiver Working Characteristics (ROC) is used to verify that the suggested design is exhibited. In addition, the proposed design is contrasted with other models that are already in use, for instance, by predicting the various exhibition measurements, as shown in Table 1. Last but not least, the computational complexity of various enhancement calculations used for tweaking the hyperparameters is determined.

Table 2 Comparative analysis of various IDS in terms of detection time of attacks

Sl.no.	Algorithm	Detection time (s)		
		80:20	70:30	50:50
1	SVM	9.4	12.3	14.3
2	CART	10.4	13.0	15.3
3	ELM	11	13.2	15.3
4	BAT-ELM	9.1	9.1	8.9
5	PLO-ELM	7.6	7.6	7.5

The developed algorithm's ROC characteristics and Confusion Matrix are shown in Figs. 2, 3 and 4 for various training data set proportions. The figures show that the proposed algorithm's performance has remained consistent despite the rise in malicious nodes in the cloud network. We compared the suggested IDS to other state-of-the-art existing IDS at various training data set proportions to demonstrate its superiority.

Figures 5, 6 and 7 display a comparative examination of several algorithms along with a variable data set comparison. From the figures, it is evident that proposed IDS has shown the constant performance even number of the malicious nodes increases, whereas the other machine learning-based IDS has shown the degraded performance as the malicious node increases. Since there are more malicious nodes now, it is obvious that the proposed IDS performs better than other learning-based IDSs.

In addition, using various ratios of training and test data, we compared the proposed IDS's detection time to that of other current IDS. The comparison of the various algorithms at various data set proportions is shown in Table 2.

From Table 2, there is no doubt that the suggested algorithm has maintained the constant detection time with the increased number of attacks in the data sets.

Conclusion Along with Future Scope

This investigation provides a unique IDS system based on the Panthera Leo Optimized Extreme Feedforward Network. The IDS introduced here in the study is used to predict and profile the common characteristics of cloud networks to overcome the ebb and flow frameworks limitations previously discussed. In addition, as the attacks grew in scope, the proposed IDS predicted numerous assaults. CIDCC data sets-2019 were considered and looked at so as to evaluate

the presentation of the suggested IDS. Despite the network's harmful hub count increasing proportionately, the recommended design's presentation resulted in high prediction accuracy of 96.5%. This demonstrates how the developed design in comparison to the other current IDS, fared better execution and faster recognition. Future IDS performance in categorizing more invisible attacks may be improved by the use of deep learning and reinforcement training techniques.

Funding No funding received for this research.

Data availability Not applicable.

Declarations

Conflict of interests No conflict of interest.

References

1. Shawish A, Salama M. Cloud computing: paradigms and technologies. In: Inter-cooperative collective intelligence: techniques andapplications. Berlin: Springer; 2014. p. 39–67.
2. El Kafhali S, Salah K. Stochastic modelling and analysis of cloud-computing data center. In: 20th Conference on Innovations in Clouds, Internet and Networks (ICIN). IEEE, 2017, pp. 122–126.
3. Anthi E, Javed A, Rana O, Theodorakopoulos G. Secure data sharing and analysis in cloud-based energy management systems. In: Cloud infrastructures services and IoT systems for smart cities. Berlin: Springer; 2017. p. 228–42.
4. Cyber hackers can now harm human life through smart meters—smart grid awareness. https://smartgridawareness.org/2014/12/30/hackers-can-now-harm-human-life/. Accessed on 02 May 2018.
5. Xiao L, Wan X, Lu X, Zhang Y, Wu D. IOT security techniques based on machine learning. arXiv preprint. arXiv:1801.06275, 2018.
6. Anthi E, Ahmad S, Rana O, Theodorakopoulos G, Burnap P. Eclipseiot: a secure and adaptive hub for the internet of things. Comput Secur. 2018;78:477–90.
7. Yu T, Sekar V, Seshan S, Agarwal Y, Xu C. Handling a trillion (unfixable) flaws on a billion devices: rethinking network security for the internet-of-things. In: Proceedings of the 14th ACM Workshop on Hot Topics in Networks, p 5. ACM, 2015.
8. Vogler M, Schleicher J, Inzinger C, Nastic S, Sehic S, Dustdar S. Leonore–large-scale provisioning of resource-constrained iot deployments. In: Service-Oriented System Engineering (SOSE), 2015 IEEE Symposium on, pp. 78–87. IEEE, 2015.
9. Gartner says 6.4 billion connected" things" will be in use in 2016, up 30 percent from 2015. https://www.gartner.com/newsroom/id/3165317. Accessed on 13 Jul 2018.
10. Gubbi J, Buyya R, Marusic S, Palaniswami M. Internet of things (iot): a vision, architectural elements, and future directions. Future Gen Comput Syst. 2013;29(7):1645–60.
11. Roesch M, et al. Snort: Lightweight intrusion detection for networks. In Lisa. 2019;99:229–38.
12. Midi D, Rullo A, Mudgerikar A, Bertino E. Kalisa system for knowledge-driven adaptable intrusion detection for the internet of things. In: Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on, pp 656–666. IEEE, 2017.
13. Bogaz Zarpelao B, Sanches Miani R, Kawakani CT, Carlistode Alvarenga S. A survey of intrusion detection in internet of things. J Netw Comput Appl. 2017;84:25–37.
14. Ojugo AA, Eboka AO, Okonta OE, Yoro RE, Aghware FO. Genetic algorithm rule-based intrusion detection system (GAIDS). 2012;3.
15. Goeschel K. Reducing false positives in intrusion detection systems using data-mining techniques utilizing support vector machines, decision trees, and naive Bayes for off-line analysis. In: Proceedings of the SoutheastCon 2016, Norfolk, VA, USA, 30 March–3 April 2016; pp. 1–6. 48.
16. Kuttranont P, Boonprakob K, Phaudphut C, Permpol S, Aimtongkhamand P, KoKaew U, Waikham B, So-In C. Parallel KNN and neighborhood classification implementations on GPU for network intrusion detection. J Telecommun Electron Comput Eng (JTEC). 2017;9:29–33.
17. Zhang B, Yu Y, Li J. Network intrusion detection based on stacked sparse autoencoder and binary tree ensemble method. In: Proceedings of the 2018 IEEE International Conference on Communications Workshops (ICC Workshops), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.
18. Gautam RKS, Doegar EA. An ensemble approach for intrusion detection system using machine learning algorithms. In: International Conference on Cloud Computing, Data Science & Engineering (Confluence): 2018;14–15.
19. Jan SU, Ahmed S, Shakhov V, Insookoo. Towards a Lightweight Intrusion Detection System for the Internet of Things. IEEE Access. 2019;7(1):42450–71.
20. Hasan M, Islam MM, Zarif MII, Hashem MMA. Attack and anomaly detection in IoT sensors in IoT sit es using machine learning approaches. Internet Things. 2019;7(1): 100059.
21. Huang G-B, Zhu Q-Y, Siew C-K. Extreme learning machine: theory and applications. Neurocomputing. 2006;70(1):489–501.
22. Wang B, Huang S, Qiu J, et al. Parallel online sequential extreme learning machine based on MapReduce. Neurocomputing. 2015;149:224–32.

Ensembling of Attention-based Recurrent Units for Detection and Mitigation of Multiple Attacks in Cloud

Kalaivani M[1], Padmavathi G[2]
Ph.D. Scholar, Department of Computer Science,
Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, India.[1]
Dean-School of Physical Sciences and Computational Sciences and Professor in the Department of Computer Science,
Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, India[2]

Abstract*—In the recent years, number of threats to network security increases exponentially as the Internet users which poses serious threat in cloud storage application. Detection and defending against the multiple threats are currently a hot topic in industry and considered as one of the challenging research in academia. Many methodologies and algorithms devised to predict the different attacks. Still, most of the methods cannot simultaneously achieve high performance of prediction with a small number of false alarm rates. In this scenario, Deep Learning (DL) algorithms are appropriate and intelligent to categorize the multiple attacks. Still, most of the existing DL techniques are computationally inefficient that may degrade the performance in predicting the both normal and attack information. To overcome this aforementioned problem, this paper proposes the hybrid combination of attention maps with deep recurrent networks to mitigate the multiple attacks with low computational overhead. Initially, the pre-processing step is proposed to the inputs in a specified range. Later on, input data are fed into the Attention Enabled Gated Recurrent Networks (AEGRN) which is used to remove the redundant features and select the optimal features that aids for the better classification. Further to enhance the faster response, deep feed forward layers are proposed to replace the traditional deep neural networks. Numerous metrics for performance, including accuracy, precision, recall, specificity, and F1-score, are examined and analyzed as part of the thorough experimentation utilizing multiple datasets, including NSL-KDD-99, UNSW -2019, and CIDC-001. Comparisons of performance between the method that is suggested and existing models developed with DL are used to demonstrate the proposed algorithm's supremacy. The suggested framework surpasses the other DL models and has the best accuracy in predicting with little computational overhead, according to an investigation.

Keywords*—Multiple threats; deep learning algorithm; attention enabled gated recurrent networks; NSL-KDD; UNSW; CIDC-001

I. INTRODUCTION

Internet Based Communication is used for managing big industry and transformed the scenario of monitoring and interaction methodology. Its scope of services also included the medical industry and was applicable to banking, schooling, government departments, military, and recreation. In addition, time, network development gives hackers and intruders opportunities to find illegal ways to break into an organization. Multiple assaults that have the capacity to deny services to legitimate customers are one of the main risks to the IP network on which numerous researchers have focused their attention. Therefore, maintaining the security and protection of various websites operating on the Internet is primarily required of the secured network [1-2]. Due to their qualities, such as quick access and primitive ways of attack detection, these attacks have expanded significantly.

It can be difficult to distinguish between malicious and lawful network data because intruders have unexpected behavior [3-5]. Applications run through anytime, anything, and anywhere in an internet context and interact remotely with a variety of devices or appliances. This makes it easier for bad actors to get devices. Despite these guidelines, interruption of devices or assistance is likely to be the first stage of many attacks due to factors such as ease of comprehension, simplicity of execution, lack of extensive technical knowledge on the part of the attacker, and variety of platforms and applications for aided attack orchestration [6–8].

These attacks can be single-source assaults commencing at just one host or multi-source attacks that distribute attack packages to the target across numerous hosts. Also, attack toolkits have been developed and therefore are easily accessible in online today [9–10]. But these tools can be exploited by the intruders to enforce the attacks with least effort. As a result, more examinations are performed in recent years through the use of numerous algorithms to develop the defensive system for cloud attacks. But these traditional systems possess the various problems such as high memory, high bandwidth and processing capacity. It is vital to design Intrusion Detection Systems in order to counteract this lack of security assaults, in which the network attacks can be prevented primarily. With exponential increase, information is steadily moved from separate networks. IDS needs improvisation in predicting the intrusion in such huge data environment.

IDS has been using deep computing and machine learning techniques in the last ten years to help classify the observed data using known characteristics or attributes that have been learned from training datasets. The purpose of ML and DL-based techniques, which have some limitations, is to evaluate network traffic packet properties and set a reasonable threshold

for separating attacks from genuine traffic. For instance, tatistical recognition methods [15], Neural Networks [11], Support Vector Machine [12], nearest neighbor [13] andclustering [14]. These current studies reveal that various studies have been conducted to offer treatments to deal with this difficulty by outlining particular treatments for emerging network assaults.

Since these intelligent-based IDS have only recently been introduced, a number of issues need to be resolved. Here are a few of the current issues that studies on attack detection systems are now facing:

1) The majority of currently used techniques concentrate on identifying a single attack with a low false alarm rate, although they typically fall short of reaching a high detection rate.

2) Knowing the many characteristics of attacks is important, but identifying the ones that can really help in the detection of assaults is even more crucial. However, because of redundant information and excessive computational expense, certain existing techniques commonly have high false positive rates. A well-organized network attack detection method remains a promising research subject because earlier methods also fall short in terms of attaining efficient accuracy.

Considering these problems, this research article proposes the novel integration of the Attention layers with the Gated recurrent NN to achieve the high classification ratio in mitigating the network attacks with less computational complexity. Following are the paper's main contributions:

1) Self-Attention Maps are introduced in Gated Recurrent Neural Network (RNN) to achieve the better feature selection that in returns support for the better detection ratio.

2) Data-Pre-processing technique is employed for the increasing the speed in detecting the attacks.

3) Feedforward Learning Layers- They are introduced in the place of the conventional neural networks to achieve the faster training with less error detection.

Following is how the manuscript is organised: Details on the background and related works are provided in Section II. The description of the dataset, data pre-processing, and suggested approach are shown in Section III. The following Section IV provides further details on the experimental findings. The Section V provides a conclusion and future enhancements.

II. History and Related Works

Abirami et al. (2022) demonstrated how "Deep Reinforcement Learning (DRL)" might be used in a cloud network to offload tasks while also recognizing generalized attackers. Techniques for identity-based linear classification are used in virtual machine attack categorization channels. This proposed system supports methods for remote information analysis. Reinforcement learning has the potential to reduce data secrecy and improve cooperation. The sole drawback of this system is the prolonged computation time [16].

In 2022, Tao et al., developed a "Continuous Duelling Deep Q-Learning (C-DDQN)" technique for protecting the cloud. The suggested Dynamic Field Adaptive System and improving are the fundamental ideas of this system. The convergence and learning capabilities of the aforementioned structure are preferable than those when transfer learning methods weren't used. But this framework's primary problem is the rising energy consumption [17].

Recurrent and convolution neural networks were combined in 2021 by Hizal et al. to create a DL method for threat detection in security of the cloud. Any discovered or forbidden traffic cannot be sent to the cloud server using this method. The recommended method is 99.86% accurate for classification into five classes. But this framework's primary drawback is the higher connectivity cost [18].

In 2020, Karri et al. proposed a three-stage abnormality detection framework that utilized DL for intrusion attack detection. CNN, GANomaly, and K-means clustering algorithms are all used by the system. The effectiveness of the network and automated intrusion detection had been either greatly improved. The main advantage of the aforementioned structure is that it reduces the level of computation without reducing cost [19].

By Wang et al. in 2022, stacking contractive auto encoder (SCAE) system was unveiled. The Support Vector Machine configuration serves as the framework's core. By using the unfiltered network information, this structure enables the automatic learning of improved as well as more trustworthy low-dimensional properties. This paradigm significantly reduces the analytical complexity. This technique leads to improved detection efficiency. This framework's drawback, nevertheless, is that it cannot be used in contexts where events happen in the present [20].

PredictDeep was introduced in 2020 as an approach for prediction of anomaly in big data environments by Elsayed et al. GCNs, or Graph Convolutional Networks, form the basis of the system. This solution produced better outcomes in regards of the fast discovery and forecasting of incidents of security and was able to cope with the multifaceted nature of clouds. The problem with this technique, though, is that it doesn't recognize and classify irregularities in a range of classifications in accordance with the changes in system function they cause [21].

Nguyen et al. (2021) examined the difficulties associated with compute offloading and cybersecurity in a multiple-user-friendly mobile edge-cloud computing framework utilizing blockchain. The above structure provides an effective authorization mechanism powered by blockchain that may protect servers in the cloud from incorrect offloading practices in order to boost offloading security. A complex DRL method called a double-dueling Q-network was developed by this framework to do this. This framework is lowering the latency, energy consumption, and intelligent contract fees. But this approach has the drawback that efficiency degrades as the amount of information increases [22].

RNN-based DL approaches were examined by Kimmel et al. (2021) for their efficacy in identifying malware in cloud.

The focus of the framework was on LSTMs and bidirectional RNNs. Such frameworks progressively understand malware behaviors based on the course of operation, minute activities, and system statistics such as CPU, memory, and disc usage. With this architecture, there are high detection rates. but, cannot maintain the identical degree of performance when dealing with diverse data [23].

Loukas et al. (2018) introduced a recurrent NN with a deep multilayer perceptron architecture which is capable of understanding the temporal context of several attacks. A computational framework was developed to determine whether compute offloading is favorable utilizing detection latency as the criterion, given networking operational parameters and DL framework processor needs. When the processing requirements are more severe and the network has become more reliable, offloading lowers detection delay to a greater extent. The biggest problem with this structure though, is the additional communication complexity [24].

By fusing a Convolutional NN with Grey Wolf Optimization, Garg et al. (2019) created a composite data mining method for identifying network abnormalities. The GWO and CNN learning procedures were improved in order to enhance the framework's abilities for initial sample creation, exploring, taking advantage of and discarding functionality. The above structure works better in terms of precision, false alarms, and recognition rate. This strategy does have a disadvantage, too, in that it increases computing difficulty [25]. Table I following provides an overview of several relevant studies.

TABLE II. LIST OF RELATED WORKS FROM LITERATURE

Author's name	Proposed methodology	Merits	Demerits
Abirami et al., (2022)	DRL	Minimises the data secrecy	Increased computational delay
Tao et al. (2022)	Continuous duelling deep Q-learning	Fast convergence	Increased energy consumption
Hizal et al., (2021)	K-means clustering, GANomaly and CNN algorithms	Reduced the computational complexity	fails to lower down on time overhead
Karri er al.,	GANomaly and CNN algorithms	Reduced the computational complexity	However, fails to reduce time overhead
Wand et al., (2022)	Stacked Contractive Autoencoder and Support Vector Machines (SVM)	Reduced the analytical overhead	Not suitable for real time environment
Elsayed et al., (2022)	Graph Convolution Networks (GCNs)	Timely detection and prediction of security breaches	It does not predict and classify anomalies under change in the system behaviour
Nguyen et al.,(2021)	Mobile edge-cloud computation offloading system	Minimized the long-term system costs of latency, energy consumption	Performance gets degraded when the data is increased
Kimmel et al.,	LSTM and Bidirectional RNNs (BIDIs)	Achieves high detection rates	Does not handle heterogeneous data
Loukas et al., (2018)	MLP and RNN	Reduction in detection latency	Increased communication overhead
Gard et al., (2019)	Grey wolf Optimization (GWO) and (CNN)	High accuracy and high detection rate	Increased computational complexity

III. PROPOSED ARCHITECTURE

According to Fig. 1, the hybrid suggested network's framework is made up of three sub modules. In the first module, multiple datasets are pre-processed and inputted to the proposed network. The second module consists of the proposed SA-GRU-FF framework in which attention layer is integrated to remove redundant and non-optimal temporal features. These features are then fed into the fully connected deep feed forward networks based on Extreme Learning Machines (ELM) for classification of the multiple attacks.

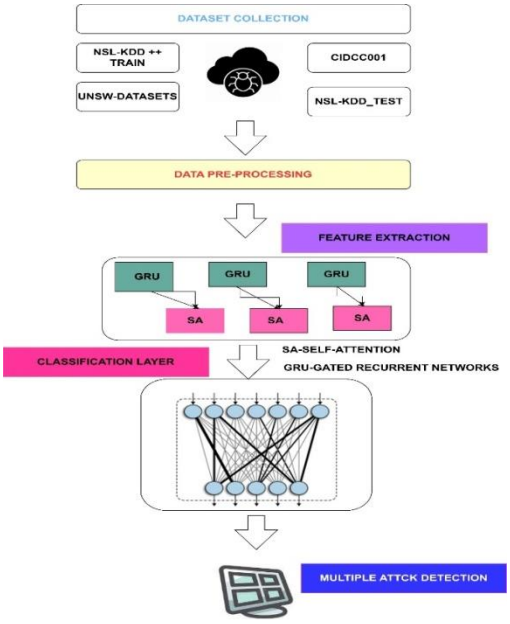


Fig. 1. Proposed architecture for the GRU-SA-FF based multiple classifications of attacks.

A. Materials and Methods

Three distinct datasets, namely CIDDS-001 [27], UNSW-NB15 [28], and NSLKDD [29], are employed in this investigation. We choose the CIDDS-001 and UNSW-NB15 datasets because they are the most current statistics produced and include real data traffic, which makes them beneficial for designing accurate IDSs for tracking and finding novel forms of denial of service attacks in cloud networks. An IDS based on anomalies may now be created with the help of the CIDDS-001 dataset, which was just made accessible. In all, the collection contains around 32 million tracks, covering both normal and attack traffic. This dataset is composed of 12 identifying features and two distinguishing traits. Random sampling is applied to acquire 80,000 normal and 20,000 DoS attack events from the relational database of server traffic data, totaling 100,000 events. Using the extracted sample, the cross-fold validity and hold-out of the classifiers are tested. A new contribution to the public domain, the UNSW-NB15 dataset,

was also utilized for the purposes of testing. In the dataset, there are 49 characteristics and 1 class attribute. A subset of the dataset uses the training and test establishes, "UNSW NB15 Train & UNSW NB15 Test". There are 175,341 occurrences in the train set compared to 82,332 in the test set. "There are 56,000 occurrences of ordinary traffic and 119,341 illustrations of attack traffic on the platform set. Additionally, there are 37,000 examples of ordinary traffic and 45,332 cases of attack traffic in the test set". Hold-out confirmation makes use of both the whole train set and the test set, while cross-fold assessment solely utilizes the set that has been tested. The NSL-KDD dataset is then utilized to do classifier validation as well. 41 measures including 1 class attribute are part of the dataset. The NSL KDD dataset's KDDTrain+ (training) as well as KDDTest+ (testing) sets are utilised in this study. 13,499 attack traffic instances and 11,743 regular traffic instances make up the total 25,192 instances in the KDDTrain+ set. While the KDDTest+ set has a total of 22,544 instances, including 12,833 instances of regular traffic and 9,711 incidents of attack traffic. On each dataset separately, hold out as well as cross fold validation of classifers are performed. The selection of these sets was made to prevent randomly selecting cases from the entire NSL-KDD dataset.

B. Data Reorganizing

The input data are first analysed, and then they are fed into a standardization approach, which assists to convert the bulk of attributes with numerical data to a specified numeric domain. Min-Max normalisation is used in conjunction with the linear transformation concept to accomplish this. After pre-processing step, new pre-processed datasets is formed from the original raw datasets. These pre-processed data is given for feature extraction module.

C. Feature Extraction using Self –Attention Gated Recurrent Networks

The operation of gated recurrent sections, self-attention, and mixed combinations of self-attention gated recurrent units are covered in this section.

1) Gated recurrent units – An overview: One of most interesting form of LSTM is known as GRU the architecture is depicted in Fig. 2. The forget gate with input vector are intended to be combined into a single vector according to the concept set out by Chung et al. [30]. Both long-term sequences and memories are supported by this network. When contrasted to the LSTM network, the complexity is drastically reduced.

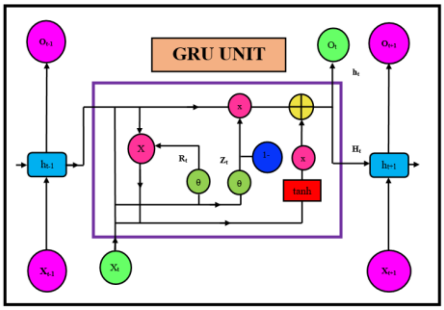


Fig. 2. GRU's architecture.

Chung developed the following equations to illustrate the traits of GRU.

$$h_t = (1 - x_t) \odot h_{t-1} + x_t \odot h_t \quad (1)$$

$$\widetilde{h_t} = g(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h \quad (2)$$

$$z_t = \sigma(W_h x_t + U_z h_{t-1} + b_z) \quad (3)$$

$$r_t = \sigma(W_h x_t + U_r h_{t-1} + b_r) \quad (4)$$

The following is the general GRU characteristic equation:

$$R = GRU(\sum_{t=1}^{n}[x_t, h_t, z_t, r_t(W(t), B(t), \eta(tannh))] \quad (5)$$

where, "xt ⇨input feature at the present state, yt ⇨output state , ht ⇨ output of the unit as of this moment, Zt & rt⇨update & reset gates, W(t) ⇨ weights, B(t) ⇨ bias weights at present instant".

2) Self-awareness maps: In 2014, the attentive map was proposed to describe the appropriate words in a sequence-to-sequence structure. In the vast mainstream of contemporary works, redundant characteristics that support accurate categorization mechanisms are imitated using attention layers. The self-attention process, commonly alluded to as the intra-attention procedure, generates the three vectors Q, K, and V for each input pattern. Thus, the results sequences are created by transforming the input patterns from all of the layers. It is a technique that, in its simplest form, maps the query string to the set of key-pair collections using logarithmic dot processes. The mathematical formula that follows can be used to get the dot multiplying for self-attention.

$$F(K, Q) = ((K), Q^T))/(V_K)\text{^}0.5 \quad (6)$$

D. Proposed Feature Extraction

BiGRU networks, which combine forward and backward GRU, are built for gathering meaningful information from the many dataset streams. Eq. (9) delivers data on the precise properties of the BiGRU network. In order to classify data, the BiGRU network collects spatiotemporal characteristics that incorporate a variety of different pieces of data. Although the training time, which makes up the overhead in the classification layer, may be affected by the more varied information in these characteristics. Self-attention layers that are inserted among the BiGRU network and classification layer help to diminish the resulting classification cost. Eq. (6) is used to create the attention characteristics retrieved from the input features of the BiGRU network that are then given to the feed-forward level of classification via the softmax layer.

$$P(F) = GRU\left(\sum_{t=1}^{n}[x_t, h_t, z_t, r_t(W(t), B(t), \eta(tannh))\right] \quad (7)$$

$$P(B) = GRU\left(\sum_{t=1}^{n}[x_t, h_t, z_t, r_t(W(t), B(t), \eta(tannh))\right] \quad (8)$$

Combining the Eq. (7) and Eq. (8)

$$P(BiGRU) = P(F) + P(B) \quad (9)$$

The following information is related to integrated Self-Attention (SA) with BiGRU feature extraction.

$$Y = Softmax\ (P(BiGRU), F(K, Q)) \quad (10)$$

E. Feed Forward Classification Layers

After receiving these attributes for the fully connected forward feed-forward network, the final classifying is carried out. Layers are entirely linked using the ELM principle. The principle of auto-tuning capacity underlies the operation of a particular class of neural network known as an ELM, which only uses one hidden unit. In regards to dependability, speed, and computational burden, ELM performed better than other learning models like "Support vector machines (SVM), Bayesian Classifier (BC), K-Nearest Neighbourhood (KNN), and even Random Forest".

There is just one hidden layer in this specific neural network; therefore it may not require to be modified. Compared to other learning algorithms like Random Forest and Support Vector Machines, ELM operates better, more quickly, and with lower computational cost. Small training error and improved approximation are the ELM's main benefits. ELM uses non-zero activation functions and weight biases that are automatically tuned. The ELM's intricate operating mechanism is covered in [26]. Following Attention maps, the ELM's input features maps are represented by:

$$X = F(Y) \qquad (11)$$

where, Y ⇨ features from Self Attention BiGRU network ,

The ELM's output function is represented by the symbol

$$Y(n) = X(n)\beta = X(n)X^T(\frac{1}{C}XX^T)^{-1}O \qquad (12)$$

ELM's comprehensive training is provided by:

$$S = \alpha(\sum_{n=1}^{N}(Y(n), B(n), W(n)) \qquad (13)$$

Finally, the softmax activation layers are applied for the above feedforward layers to achieve the best accuracy.

IV. Experimentation Details

The entire algorithm was designed on an Intel Workspace with a 3.2 GHz of frequency, I7 CPU (NVIDIA GPU) and a16GB of RAM. Utilizing Keras (Tensorflow) as the rear end, the suggested baseline infrastructure was created.

A. Performance Metrics

Deep feed forward training networks that classify the necessary classifications into typical sensitive and malicious information as well as the suggested design are validated as part of the experiment. Metrics including "accuracy, sensitivity, selectivity, recall, and f1-score" are used to gauge the suggested design's effectiveness. The calculations for the metrics used to assess the suggested architecture are shown in Table II in their respective computation formulae. Additionally, Table III shows the experimental hyperparameters that were utilized to train the suggested network.

B. Results and Discussion

The experimentation is carried out based on component structures with the same parameters as the proposed framework. In detail, the existing structures were one dimensional Long Short Term Memory [30], Gated Recurrent Units [31], Optimized LSTM [23], and BiGRU [32]. The technique was validated and a comparison study was performed using four different datasets.

TABLE III. Algebraic Equations for the Calculation of Performance Metrics

SL. NO	Validation Metrics	Formulae
01	Accuracy (A_{cc})	$\frac{TP+TN}{TP+TN+FP+FN}$
02	Sensitivity or recall (R_{ll})	$\frac{TP}{TP+FN}$x100
03	Specificity (S_{ty})	$\frac{TN}{TN+FP}$
04	Precision (P_{cn})	$\frac{TN}{TP+FP}$
05	F1-Score (F_{cr})	$\frac{Precision * Recall1}{Precision + Recall1}$

Where, "TP – True Positive Values, TN – True Negative Values, FP – False Positive and FN – False Negative"

TABLE IV. Hyper Parameters used in the Network's Training

SL. NO	Hyper-Parameters	Specifications
1	GRU cell count	10
2	Epochs count	200
3	Batch Size	30
4	Learning Rate	0.001
5	Momentum	0.2
6	Dropouts	0.2

TABLE V. Using the CIDCC-001 Datasets, Efficiency Statistics of the Distinct Algorithms

Algorithms	Validation Metrics				
	A_{cc}	P_{cn}	R_{ll}	S_{ty}	F_{cr}
LSTM	0.89	0.85	0.834	0.190	0.84
GRU	0.91	0.86	0.856	0.1556	0.857
Optimized-GRU	0.92	0.89	0.887	0.1290	0.885
Proposed Model	0.98	0.97	0.966	0.11	0.975

TABLE VI. Using UNSW2019 Datasets, Monitoring of the Multiple Algorithms

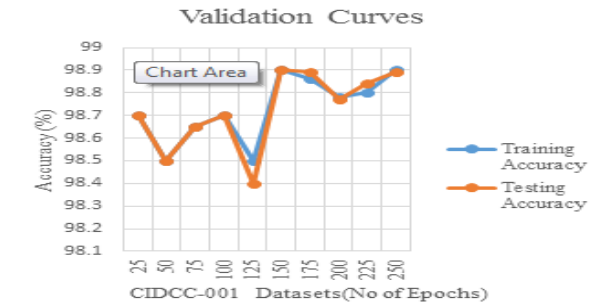
Algorithms	Validation Metrics				
	A_{cc}	P_{cn}	R_{ll}	S_{ty}	F_{cr}
LSTM	0.874	0.87	0.864	0.150	0.86
GRU	0.902	0.90	0.89	0.110	0.91
Optimized-GRU	0.910	0.91	0.90	0.100	0.92
Proposed Model	0.983	0.98	0.974	0.011	0.983

TABLE VII. NSL-KDD+(TRAIN) DATASETS PERFORMANCE INDICATORS OF THE SEVERAL ALGORITHMS

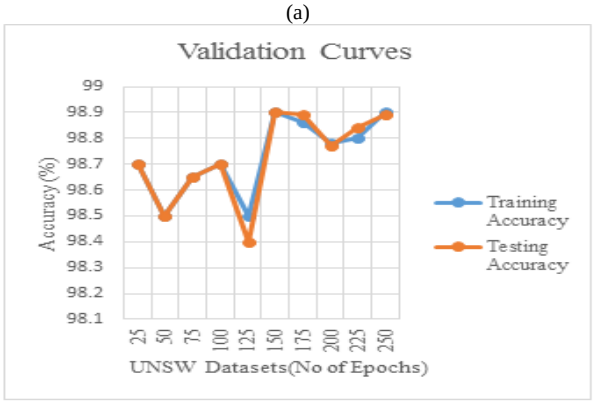
Algorithms	Validation Metrics				
	A_{cc}	P_{cn}	R_{ll}	S_{ty}	F_{cr}
LSTM	0.88	0.875	0.834	0.190	0.84
GRU	0.92	0.90	0.856	0.1556	0.857
Optimized-GRU	0.93	0.92	0.887	0.1290	0.885
Proposed Model	0.988	0.98	0.974	0.001	0.980

TABLE VIII. NSL-KDD+(TEST) DATASETS PERFORMANCE METRICS FOR THE DIFFERENT ALGORITHMS

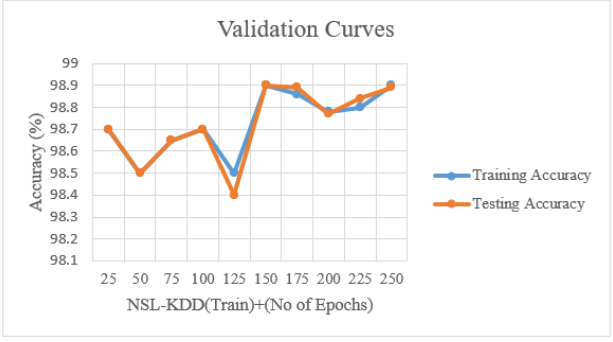
Algorithms	Validation Metrics				
	A_{cc}	P_{cn}	R_{ll}	S_{ty}	F_{cr}
LSTM	0.88	0.875	0.834	0.190	0.84
GRU	0.92	0.90	0.856	0.1556	0.857
Optimized-GRU	0.93	0.92	0.887	0.1290	0.885
Proposed Model	0.988	0.98	0.974	0.001	0.980



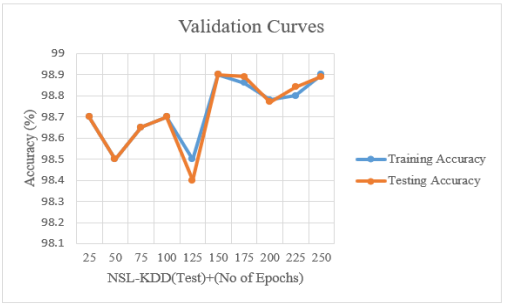
(a)



(b)



(c)



(d)

Fig. 3. Validation performance of the suggested model using distinctive datasets a) CIDCC-001 datasets b) UNSW-datasets c) NSL-KDD datasets (Train) d) NSL-KDD datasets (Test).

Tables IV, V, VI, and VII demonstrates the proposed algorithm's performance metrics for categorizing several assaults using various datasets. The Table IV represents, the outcomes of proposed and existing frameworks when testing under CIDCC-001 Datasets. The Table V, Table VI and Table VI represents, the outcomes of proposed and existing model when testing under UNSW2019, NSL-KDD+(Train) and NSL-KDD+(TEST) datasets respectively. From Table IV, V, VI and VII, it is observed that, the suggested model GRU-SA-FF has demonstrated the best performance in detecting the numerous attacks. The integration of Self-attention maps has provided the best results in contrast to different DL techniques. Additionally, the validation effectiveness of the suggested model (see Fig. 3) is assessed using various datasets, and it is discovered that the RMSE (root mean square error) in between training and testing data is 0.001.

TABLE IX. MBT IN SUPPORT OF DIFFERENT ALGORITHMS USING DIFFERENT DATASETS

Datasets	(MBT)-secs			
	LSTM	GRU	Op-LSTM	Proposed Model
CIDCC001	0.5	0.45	0.37	0.23
UNSW	0.45	0.39	0.31	0.21
NSL-KDD++ Train	0.5	0.45	0.37	0.22
NSL-KDD++ Test	0.43	0.42	0.38	0.22

Model building times for various classifiers are shown in Table VIII for four datasets employing hold-out evaluation. Recognising how essential it is to deliberate how long a system needs train until it is successful at spotting various risks, the main driver aimed at estimating MBT is this realisation. Because of this, MBT helps to achieve a good trade-off among computational complexity and the accuracy of classifiers. The suggested model's average MBT when trained on the different sets of data is 0.22s, according to the above table, compared to 0.36s, 0.41s, and 0.48s for Op-LSTM, GRU, and LSTM, respectively, for Op-LSTM. According to the evaluation, the suggested framework uses only 0.22 seconds and excels at designing countermeasures against several threats.

V. CONCLUSION AND FUTURE ENHANCEMENT

In this work, investigation on integration of Self-attention maps with GRU for securing the cloud against the multiple attacks is carried out. The role of self –attention network with the BiGRU to select the optimal features that can aid for the classification layers is proposed in this paper. Additionally,

role of feed forward layers which works on principle of ELM has been used in the proposed research to achieve the better classification with reduced computational burden and quick speed. Precision, specificity, susceptibility, false alarm rate, and region under the curve of receiver operating characteristics are used to assess the performance of the suggested model. On the CIDDS-001, UNSW-NB15, & NSL-KDD datasets, all of the classifiers are benchmarked. Results demonstrate in terms of a superior detection ratio and so little overhead, the proposed approach have done better over the other DL models. As the future scope, performance of the proposed model is required for the validation with real time datasets and also brighter light of deploying in the resource constraint in Cloud.

REFERENCES

[1] Laghrissi, F., Douzi, S., Douzi, K. et al., "IDS-attention: an efficient algorithm for intrusion detection systems using attention mechanism." J Big Data, Vol 8, 149, 2021.

[2] Maha M, Althobaiti K, Mohan KP, Deepak G, Sachin K, Mansour RF. "An intelligent cognitive computing based intrusion detection for industrial cyber-physical systems." Measurement. 2021, 186(110145):0263–2241.

[3] Anthi E, Javed A, Rana O, Theodorakopoulos G "Secure data sharing and analysis in cloud-based energy management systems." In Cloud Infrastructures, Services, and IoT Systems for Smart Cities, pages 228–242. Springer, 2017

[4] Baykara, M., & Das, R. "A novel hybrid approach for detection of webbased attacks in intrusion detection systems." International Journal of Computer Networks and Applications, 4(2), 62–76, 2017

[5] Bergstra, J., &Bengio, Y. (2012). "Random search for hyper-parameter optimization." Journal of Machine Learning Research, 13(Feb), 281–305.

[6] Mahboob AS, Moghaddam MRO. "An Anomaly-based Intrusion Detection System Using Butterfy Optimization Algorithm," 6th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS), 2020; pp. 1-6

[7] BButun, I., Morgera, S. D., &Sankar, R.. "A survey of intrusion detection systems in wireless sensor networks." IEEE Communications Surveys & Tutorials, 16(1), 266–282, 2014

[8] Chen, T., &Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In ACM, proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 785–794).

[9] Khan MA. HCRNNIDS: hybrid convolutional recurrent neural network-based network intrusion detection system. Processes. 2021; 9(5): 834.

[10] Shen Y, Zheng K, Wu C, Zhang M, Niu X, Yang Y. An ensemble method based on selection using bat algorithm for intrusion detection. Comput J. 2018;61(4):526–38.

[11] Demšar, J. (2016). Statistical comparisons of classifers over multiple data sets. Journal of Machine Learning Research, 7(Jan), 1–30.

[12] Dhanjani, N. (2013). Hacking lightbulbs: Security evaluation of the philips hue personal wireless lighting system. Retrieved November 3, 2019, from https://www.dhanjani.com/docs/Hacking

[13] Diro, A. A., &Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for Internet of Things. Future Generation Computer Systems, 82, 761–768.

[14] Girma A, Garuba M, Goel R. Advanced machine language approach to detect DDoS attack using DBSCAN clustering technology with entropy. In: Latifi S, ed. Information Technology - New Generations. Advances in Intelligent Systems and Computing, 2018, vol. 558. Cham, Switzerland: Springer, pp. 125–131

[15] Douglas, P. K., Harris, S., Yuille, A., & Cohen, M. S. (2011). Performance comparison of machine learning algorithms and number of independent components used in FMRI decoding of belief vs. disbelief. Neuroimage, 56(2), 544–553.

[16] P. Abirami, S. Vijay Bhanu and T. K. Thivakaran, "Crypto-Deep Reinforcement Learning Based Cloud Security for Trusted Communication," 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), 2022, pp. 1-10, doi: 10.1109/ICSSIT53264.2022.9716429.

[17] Y. Tao, J. Qiu and S. Lai, "A Hybrid Cloud and Edge Control Strategy for Demand Responses Using Deep Reinforcement Learning and Transfer Learning," in IEEE Transactions on Cloud Computing, vol. 10, no. 1, pp. 56-71, 1 Jan.-March 2022, doi: 10.1109/TCC.2021.3117580.

[18] S. Hizal, Ü. ÇAVUŞOĞLU and D. AKGÜN, "A new Deep Learning Based Intrusion Detection System for Cloud Security," 2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), 2021, pp. 1-4, doi: 10.1109/HORA52670.2021.9461285.

[19] C. Karri and M. S. R. Naidu, "Deep Learning Algorithms for Secure Robot Face Recognition in Cloud Environments," 2020 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom), 2020, pp. 1021-1028, doi: 10.1109/ISPA-BDCloud-SocialCom-SustainCom51426.2020.00154.

[20] W. Wang, X. Du, D. Shan, R. Qin and N. Wang, "Cloud Intrusion Detection Method Based on Stacked Contractive Auto-Encoder and Support Vector Machine," in IEEE Transactions on Cloud Computing, vol. 10, no. 3, pp. 1634-1646, 1 July-Sept. 2022, doi: 10.1109/TCC.2020.3001017.

[21] M. A. Elsayed and M. Zulkernine, "PredictDeep: Security Analytics as a Service for Anomaly Detection and Prediction," in IEEE Access, vol. 8, pp. 45184-45197, 2020, doi: 10.1109/CCESS.2020.2977325.

[22] D. C. Nguyen, P. N. Pathirana, M. Ding and A. Seneviratne, "Secure Computation Offloading in Blockchain Based IoT Networks With Deep Reinforcement Learning," in IEEE Transactions on Network Science and Engineering, vol. 8, no. 4, pp. 3192-3208, 1 Oct.-Dec. 2021, doi: 10.1109/TNSE.2021.3106956.

[23] J. C. Kimmel, A. D. Mcdole, M. Abdelsalam, M. Gupta and R. Sandhu, "Recurrent Neural Networks Based Online Behavioural Malware Detection Techniques for Cloud Infrastructure," in IEEE Access, vol. 9, pp. 68066-68080, 2021, doi: 10.1109/ACCESS.2021.3077498.

[24] G. Loukas, T. Vuong, R. Heartfield, G. Sakellari, Y. Yoon and D. Gan, "Cloud-Based Cyber-Physical Intrusion Detection for Vehicles Using Deep Learning," in IEEE Access, vol. 6, pp. 3491-3508, 2018, doi: 10.1109/ACCESS.2017.2782159.

[25] S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A. Y. Zomaya and R. Ranjan, "A Hybrid Deep Learning-Based Model for Anomaly Detection in Cloud Datacenter Networks," in IEEE Transactions on Network and Service Management, vol. 16, no. 3, pp. 924-935, Sept. 2019, doi: 10.1109/TNSM.2019.2927886.

[26] Verma, A., &Ranga, V. (2018). Statistical analysis of CIDDS-001 dataset for network intrusion detection systems using distance-based machine learning. Procedia Computer Science, 125, 709–716.

[27] Verma, A., &Ranga, V. (2019a). ELNIDS: Ensemble learning based network intrusion detection system for RPL based Internet of Things. In 2019 4th International conference on Internet of Things: Smart innovation and usages (IoT-SIU) (pp. 1–6). IEEE.

[28] Verma, A., &Ranga, V. (2019). Evaluation of network intrusion detection systems for RPL based 6LoWPAN networks in IoT. Wireless Personal Communications, 108(3), 1571–1594.

[29] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," arXiv preprint arXiv:1412.3555, 2014

[30] Staudemeyer RC. 1 Applying long short-term memory recurrent neural networks to intrusion detection. South AfrComput J. 2015;56(1):136–54.

[31] Kim J, Kim J, Thu HLT, and Kim H. Long short term memory recurrent neural network classifer for intrusion detection, In Proc. Int. Conf. Platform Technol. Service (PlatCon); 2016, pp. 1–5.

[32] Shen Y, Zheng K, Wu C, Zhang M, Niu X, Yang Y. An ensemble method based on selection using bat algorithm for intrusion detection. Comput J. 2018;61(4):526