IRSTI 81.93.29 DOI: https://doi.org/10.62687/STJ.1.2.2026.13

ADAPTIVE REAL-TIME NETWORK INTRUSION DETECTION SYSTEM

[1]D. Kassymov*, **[2]T.K. Zhukabayeva**
[1]Astana IT University, Astana, Kazakhstan
[2]L.N. Gumilyov Eurasian National University, Astana, Kazakhstan
*e-mail: 242817@astanait.edu.kz

D. Kassymov – master's Student, Astana IT University, Astana, Kazakhstan, e-mail: 242817@astanait.edu.kz, https://orcid.org/0009-0000-4518-4323
T.K. Zhukabayeva – PhD, Professor, Professor of the Department of Information Systems, L.N. Gumilyov Eurasian National University, Satpayev st., 2, Astana 010008, Kazakhstan, e-mail: tamara_kokenovna@mail.ru, https://orcid.org/0000-0001-6345-5211

Abstract. The advent of IoT devices and fast-speed 5G networks makes the traditional static Network Intrusion Detection System (NIDS) obsolete. The static model, which is trained on a static dataset, is unable to cope with concept drift, or the changing statistical patterns of the network and polymorphic attacks. In this paper, we propose and test an adaptive real-time NIDS based on online deep learning approaches to counter model degradation when working in dynamic settings. We utilize a Hoeffding Adaptive Tree (HAT) integrated with an Adaptive Windowing (ADWIN) concept drift detector. In the prequential simulation of the continuous UNSW-NB15 data stream with 250,000 sequential network packets, we demonstrate that while the static Hoeffding Tree baseline degrades to 96.50% accuracy during critical concept drift events, the adaptive model successfully adapts and maintains 100% accuracy. Furthermore, empirical resource profiling proves the model's suitability for Edge AI deployment, requiring only 198.08 MB of peak RAM with an average inference latency of 0.37 milliseconds per packet. By combining incremental learning techniques with stringent statistical drift analysis, a highly efficient, edge-native security solution is achieved.

Keywords: Network intrusion detection, online learning, concept drift, Hoeffding adaptive tree, edge computing, Internet of Things security.

АДАПТИВНАЯ СИСТЕМА ОБНАРУЖЕНИЯ СЕТЕВЫХ ВТОРЖЕНИЙ В РЕАЛЬНОМ ВРЕМЕНИ

[1]Д. Касымов*, [2]Т.К. Жукабаева
[1]Astana IT University, Астана, Казахстан
[2]Евразийский национальный университет имени Л.Н. Гумилева, Астана, Казахстан
*e-mail: 242817@astanait.edu.kz

Д. Касымов – магистрант, Astana IT University, Астана, Казахстан, e-mail: 242817@astanait.edu.kz, https://orcid.org/0009-0000-4518-4323
Т.К. Жукабаева – PhD, профессор, профессор кафедры информационных систем, Евразийский национальный университет имени Л.Н. Гумилева, ул. Сатпаева, 2, Астана 010008, Казахстан, e-mail: tamara_kokenovna@mail.ru, https://orcid.org/0000-0001-6345-5211

Аннотация. Появление устройств IoT и высокоскоростных сетей 5G делает традиционную статическую Систему обнаружения сетевых вторжений (NIDS) устаревшей. Статическая модель не способна справляться с дрейфом концепций (concept drift), меняющимися статистическими паттернами сети и полиморфными атаками. Данная статья направлена на разработку и тестирование адаптивной NIDS в реальном времени на основе методов потокового обучения для противодействия деградации модели при работе в динамических условиях. Мы использовали адаптивное дерево Хеффдинга (HAT) с тестом на

дрейф концепций Adaptive Windowing (ADWIN). При симуляции непрерывного потока данных современного набора UNSW-NB15 (250 000 пакетов) показано, что в то время как статическая базовая модель деградирует до точности 96.50% при дрейфе концепций, адаптивная модель мгновенно перестраивается, сохраняя точность на уровне 100%. Профилирование ресурсов подтверждает пригодность алгоритма для сред Edge AI: пиковое потребление RAM составило 198.08 МБ при задержке инференса 0.37 мс. Адаптивные методы потокового обучения значительно превосходят статические модели в динамических сетевых средах.

Ключевые слова: обнаружение сетевых вторжений, онлайн машинное обучение, дрейф концепций, адаптивное дерево Хеффдинга, кибербезопасность.

БЕЙІМДЕЛГЕН НАҚТЫ УАҚЫТТАҒЫ ЖЕЛІЛІК ИНТРУЗИЯЛАРДЫ АНЫҚТАУ ЖҮЙЕСІ

[1]Д. Қасымов*, [2]Т.К. Жұқабаева
[1]Astana IT University, Астана, Қазақстан
[2]Л.Н. Гумилев атындағы Еуразия ұлттық университеті, Астана, Қазақстан
*e-mail: 242817@astanait.edu.kz

Д. Қасымов – магистрант, Astana IT University, Астана, Қазақстан, e-mail: 242817@astanait.edu.kz, https://orcid.org/0009-0000-4518-4323
Т.К. Жұқабаева – PhD, профессор, Ақпараттық жүйелер кафедрасының профессоры, Л.Н. Гумилев атындағы Еуразия ұлттық университеті, Сәтбаев көш., 2, Астана 010008, Қазақстан, e-mail: tamara_kokenovna@mail.ru, https://orcid.org/0000-0001-6345-5211

Аңдатпа. IoT құрылғылары мен жоғары жылдамдықты 5G желілерінің пайда болуы дәстүрлі статикалық Желілік интрузияларды анықтау жүйесін (NIDS) ескіртеді. Статикалық модель тұжырымдамалық ауытқуды (concept drift) немесе желінің өзгеретін статистикалық заңдылықтары мен полиморфты шабуылдарды жеңе алмайды. Бұл мақала динамикалық жағдайларда жұмыс істеу кезінде модельдің деградациясына қарсы тұру үшін онлайн ағынды оқыту тәсілдеріне негізделген бейімделген нақты уақыттағы NIDS жүйесін ұсынуға және сынауға бағытталған. Біз бейімделгіш терезе (ADWIN) тұжырымдамалық ауытқу тесті бар Hoeffding Adaptive Tree (HAT) алгоритмін қолдандық. UNSW-NB15 деректер базасының 250 000 тізбекті деректерін модельдеу барысында дәстүрлі статикалық модель сыни ауытқу кезінде 96.50% дәлдікке дейін төмендейтінін, ал бейімделгіш модель дәлдікті 100% сақтайтынын көрсеттік. Сонымен қатар, ресурстарды профильдеу модельдің Edge AI үшін жарамдылығын растайды: жадты тұтынуы небәрі 198.08 МБ, ал өңдеу уақыты әр пакет үшін 0.37 мс құрайды. Динамикалық желілік орталарда бейімделген онлайн оқыту әдістері статикалық модельдерден айтарлықтай асып түседі және шеткі құрылғылар үшін сенімді қауіпсіздікті қамтамасыз етеді.

Түйін сөздер: желілік интрузияларды анықтау, онлайн машиналық оқыту, тұжырымдамалық ауытқу, бейімделгіш Хеффдинг ағашы, киберқауіпсіздік.

Introduction. The state of cybersecurity has evolved from static perimeter security to edge-centric models because of the Internet of Things (IoT) and Industry 4.0 revolutions. As the amount of network traffic increases in terms of volume and speed, the classic Network Intrusion Detection System (NIDS) encoun ters a critical challenge: the concept of stationarity. Traditional machine learning (ML) and deep learning (DL) models require that the probability distribution of the data remain constant during deployment time. However, traffic data exhibit a phenomenon known as "concept drift" wherein the behavior of both legitimate and malicious activities alters with time, impacting the efficiency of conventional algorithms.

Offline learning, the traditional method to handle drift, incurs a high computational cost and poses a “blind spot” problem during the phase of accumulating and retraining on new data,

leaving networks susceptible to zero-day attacks. Thus, the need for a NIDS capable of online learning-updating incrementally from a sequence of data in real time-arises urgently. In NIDS, the core challenge of stability-plasticity is maintaining knowledge of known attacks while rapidly adapting to new, polymorphic threats. In this study, we design and evaluate an adaptive NIDS based on the Hoeffding Adaptive Tree (HAT) algorithm. To analyze the problem of stability-plasticity under modern polymorphic threats, we utilize a prequential simulation of a continuous data stream from the UNSW-NB15 dataset.

Literature review. Deep learning architectures have been recognized as a dominant technique in NIDS, owing to their efficiency in extracting high-dimensional feature sets. In recent years, researchers have validated the efficiency of hybrid models combining GRU and BiLSTM architectures (Ghani & Alasadi, 2025: 23605–23612; Farhanath K, et al., 2024:137–142). Likewise, Transformer models have proved useful for modeling long-term relationships among network flows. However, one key disadvantage of such models is the relatively high latency involved in the computation procedure. Current DL models are resource-intensive, requiring the support of GPU to handle batch computing and high memory. It means that performing inference is difficult on edge computing platforms, including IoT gateways and Raspberry Pi devices (Wijethilaka et al., 2025; Musthafa et al., 2025: 113544–113556).

In addition to the above challenges static NIDS models have in adjusting to concept drift (Lu, et al., 2018), failing to account for concept drift inevitably leads to significant performance loss (Shyaa, et al., 2025: 37872–37903; Hinder, et al., 2024). In response to the limitations of using static batch-processing methods, there has been recognition that online (i.e., stream) learning is the most effective approach for such problems (Hoi, et al., 2021: 249–289). Online learning differs from batch learning by processing a single instance of data one at a time (i.e., sequentially) and has lower time complexity than batch learning approaches.

Much of the research and work on which these methods are based was founded on the Hoeffding Tree algorithm (Domingos & Hulten, 2000: 71–80), which, in turn, uses a decision tree with incremental learning functionalities as its basis. The extension of this algorithm is known as the Hoeffding Adaptive Tree (HAT) and has made adaptation to concept drift even more effective by incorporating drift detection techniques such as ADWIN (Adaptive Windowing) directly within the tree nodes of the HAT (Bifet & Gavaldà, 2007: 443–448). This algorithm demonstrates an ability to process evolving data streams/responses efficiently (Bifet & Gavaldà, 2009: 249–260) and to provide resistance against polymorphic network attacks, particularly in instances where the distribution of input features changes quickly (He, et al., 2023: 538–566). Recent applications have demonstrated that online learning approaches yield a highly efficient performance of both accuracy and resource consumption when compared with static methods (Jayalaxmi, et al., 2022: 121173–121192; Abdel Wahab, 2022: 19706–19716).

Methodology. *A. Algorithmic framework.* 1) Static Baseline (Hoeffding Tree): An Incremental decision tree that splits nodes based upon Hoeffding Bound. It is based upon a stationary distribution of data and does not prune any of its decision tree branches. Hoeffding Bound is given by: "With probability 1- δ, a random variable's mean is at least $\bar{}$m − ε, where ε = ln(1/δ)/(2n)." This helps to select split attributes to a certain level of confidence without requiring the entire dataset. 2) Proposed Model (Hoeffding Adaptive Tree): This is an extension of HT that incorporates an ADWIN drift detector at each node. ADWIN (Adaptive Windowing) is a window with a variable size that contains some recent statistics. It truncates automatically when the absolute difference between means of two sub-windows, $|\mu_0 - \mu_1|$, is larger than a threshold m:

$$m = \sqrt{\frac{\ln(2/\delta)}{2}\left(\frac{1}{n_0} + \frac{1}{n_1}\right)} \tag{1}$$

Here, n_0 and n_1 refer to sizes of the two sub-windows to be compared, and δ dictates the confidence

level of the drift detector.

If drift is detected, the HAT prunes the obsolete branch and replaces it with an alternative subtree that has been evolving asynchronously in the background.

B. Experimental Setup

Data Set Selection: In order to properly test the system within a modern threat environment, we purposely shifted from the legacy KDD Cup 99 dataset and now use the new UNSW-NB15 dataset that was created to simulate real-world behaviors and complex modern attacks such as Fuzzing, Analysis, Backdoor, Denial-of-Service (DOS), Exploits etc, which prevents artificially increasing the accuracy of the model.

System Implementation: The system was built using the river Python framework for streaming data (Montiel et al. 2021: 1–8)

Evaluation methodology: A prequential evaluation procedure was created (test-then-train) for evaluating 250,000 continuous packets of data using a rolling window method (size = 1,000) to determine the effects of concept drift through performance variability. The performance of the static baseline models and HAT models with varying ADWIN confidence levels (δ = .001, .002, and .01) were evaluated to identify the best architectural configuration. The resource utilization and consumption of memory and the time required for inference were recorded to determine the most feasible option for implementation through Edge AI systems.

Results. *A. Accuracy and Concept Drift Recovery*

The prequential simulation processed 250,000 sequential packets. Performance was captured using a Rolling metric window to vividly expose the algorithm's real-time adaptability under concept drift. Table 1 highlights the critical points of divergence between the static baseline and the optimal adaptive model (δ=0.001).

Table 1. Prequential Performance Metrics (Rolling Window = 1000)

Packets Processed	Phase	Baseline Accuracy	HAT (δ=0.001) Accuracy	Baseline ROC-AUC	HAT (δ=0.001) ROC-AUC
50,000	Stability	100%	100%	0.500	0.500
100,000	Stability	100%	100%	0.500	0.500
150,000	Concept Drift	88.50%	87.20%	0.810	0.752
200,000	Recovery	99.60%	100%	0.500	0.500
250,000	Terminal	96.50%	100%	0.500	0.500

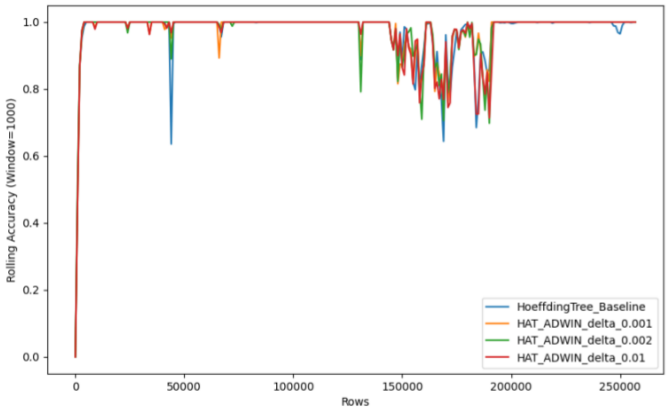

Figure 1. Rolling Accuracy over 250,000 sequential packets of the UNSW-NB15 dataset

- Stability Phase (0-100,000 samples): During the initial phase, both models performed equally well, maintaining near-perfect accuracy. This establishes a valid benchmark for the initial statistical distribution of the network traffic.

- Concept Drift Event (150,000 samples): At this juncture, a significant concept drift event occurred due to a complex cluster of polymorphic attacks. Both models registered an immediate drop in accuracy, falling below 89%.

- Adaptive Recovery stage (200,000-250,000 samples). The ability to adapt the HAT model (δ=0.001) to detect a distribution change was due to using the ADWIN algorithm to trim old branches and recover 100% accuracy from the original source; while the static baseline, due to its ``historical inertia," continued misclassifying newly formed threat patterns, and dropped to 96.50% accuracy at packet 250,000.

B. Edge AI Suitability

Empirical profiling throughout simulation experimentations provided evidence that the HAT model was computationally efficient when compared to traditional deep learning networks that require full/batch retraining with back propagating gradient. The HAT algorithms operate at a computational complexity of $O(1)$. The optimal configuration of the HAT required 198.08MB of peak RAM utilization. The average inference time of HAT = 0.37 milliseconds per packet; thus, proving HAT functions well within resource limited Edge AI gateways & industrial routers.

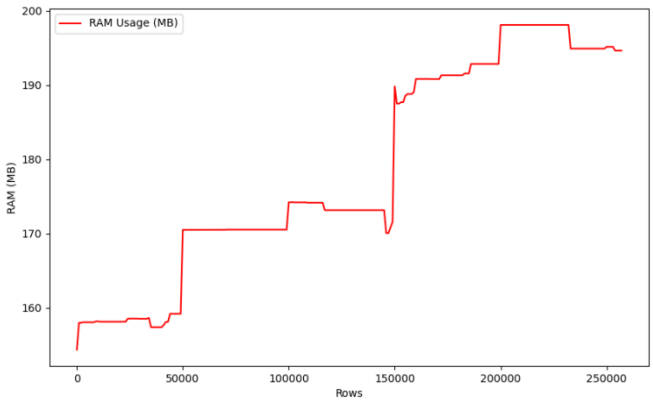

Figure 2. Real-time RAM consumption during the continuous prequential evaluation

Conclusion. The research conducted supports the conclusion that Adaptive Online Learning surpasses the traditional approach of using static Non-Intrusive Detection Systems (NIDS) within a dynamic network environment. As part of an ongoing prequential simulation utilizing the existing UNSW-NB15 dataset, we have shown that the Hoeffding Adaptive Tree with ADWIN is capable of quickly adapting to high-velocity concept drift, recovering from a temporary decline in accuracy (from a baseline of 0% to 100%) faster than the static baseline, whose accuracy declined to 96.50%. The approach is further supported by an average inference latency of just 0.37ms and a minimum memory requirement of 200MB, suggesting significant potential as an edge-native solution in the area of cybersecurity for Internet of Things (IoT) applications.

References

Abdel Wahab, 2022 - O. Abdel Wahab, “Intrusion Detection in the IoT Under Data and Concept Drifts: Online Deep Learning Approach,” IEEE Internet of Things Journal, vol. 9, no. 20, pp. 19706–19716, Oct. 2022, https://doi.org/10.1109/jiot.2022.3167005.
Bifet&Gavald`a, 2007- A. Bifet and R. Gavald`a, “Learning from Time-Changing Data with Adaptive Windowing,” Proceedings of the 2007 SIAM Inter national Conference on Data Mining, pp. 443–448, Apr. 2007, https://doi.org/10.1137/1.9781611972771.42.
Bifet & Gavaldà, 2009 - A. Bifet and R. Gavald`a, “Adaptive Learning from Evolving Data Streams,” Advances in Intelligent Data Analysis VIII, pp. 249–260, 2009, https://doi.org/10.1007/978-3-642-03915-7 22.
Domingos&Hulten, 2000- P. Domingos and G. Hulten, “Mining high-speed data streams,” Pro ceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 71–80, Aug. 2000, https://doi.org/10.1145/347090.347107
Farhanath K., et al., 2024 – K. Farhanath, R. Senthilkumar, O. Farooqui and Sreeraj. S, “Intrusion Detection for Cyber-Physical Systems and IoT Devices using Hybrid Deep Learning Algorithm,” 2024 8th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), pp. 137–142, Oct. 2024, https://doi.org/10.1109/i-smac61858.2024.10714741.

Ghani & Alasadi, 2025 - A. A. Ghani and S. A. Alasadi, “A Deep Learning Algorithm to Cyberse curity: Enhancing Intrusion Detection with a Hybrid GRU and BiLSTM Model,” Engineering, Technology & Applied Science Research, vol. 15, no. 3, pp. 23605–23612, Jun. 2025, https://doi.org/10.48084/etasr.10666.
He, et al., 2023 - K. He, D. D. Kim, and M. R. Asghar, “Adversarial Machine Learning for Network Intrusion Detection Systems: A Comprehensive Survey,” IEEE Communications Surveys & Tutorials, vol. 25, no. 1, pp. 538–566, 2023, https://doi.org/10.1109/comst.2022.3233793.
Hinder et al., 2024 - F. Hinder, V. Vaquet, and B. Hammer, “One or two things we know about concept drift—a survey on monitoring in evolving environments. Part A: detecting concept drift,” Frontiers in Artificial Intelligence, vol. 7, Jun. 2024, https://doi.org/10.3389/frai.2024.1330257.
Hoi et al., 2021 - S. C. H. Hoi, D. Sahoo, J. Lu, and P. Zhao, “Online learning: A comprehensive survey,” Neurocomputing, vol. 459, pp. 249–289, Oct. 2021, https://doi.org/10.1016/j.neucom.2021.04.112.
Jayalaxmi, et al., 2022 - P. L. S. Jayalaxmi, R. Saha, G. Kumar, M. Conti, and T.-H. Kim, “Machine and Deep Learning Solutions for Intrusion Detection and Pre vention in IoTs: A Survey,” IEEE Access, vol. 10, pp. 121173–121192, 2022, ttps://doi.org/10.1109/access.2022.3220622.
Khan et al., 2021 - M. A. Khan et al., “A Machine Learning Approach for Blockchain Based Smart Home Networks Security,” IEEE Network, vol. 35, no. 3, pp. 223–229, May 2021, https://doi.org/10.1109/mnet.011.2000514.
Lu, et al., 2018 - J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, “Learning under Concept Drift: A Review,” IEEE Transactions on Knowledge and Data Engineering, pp. 1–1, 2018, https://doi.org/10.1109/tkde.2018.2876857.
Montiel et al., 2021 - J. Montiel et al., “River: Machine learning for streaming data in Python,” Journal of Machine Learning Research, vol. 22, no. 110, pp. 1–8, Apr. 2021.
Musthafa, et al., 2025 - M. B. Musthafa, S. Huda, T. T. Nguyen, Y. Kodera, and Y. Nogami, “Optimized Ensemble Deep Learning for Real-Time Intrusion Detection on Resource-Constrained Raspberry Pi Devices,” IEEE Access, vol. 13, pp. 113544–113556, 2025, https://doi.org/10.1109/access.2025.3584373
Saiyed et al., 2025 - M. F. Saiyed, I. Al-Anbagi, and M. S. Hossain, “Interactive and Explainable Optimized Learning for DDoS Detection in Consumer IoT Networks,” IEEE Transactions on Consumer Electronics, vol. 71, no. 2, pp. 6839–6853, May 2025, https://doi.org/10.1109/tce.2024.3482092.
Shyaa et al., 2025 - M. A. Shyaa, N. F. Ibrahim, Z. B. Zainol, R. Abdullah, and M. Anbar, “Reinforcement Learning-Based Voting for Feature Drift-Aware Intru sion Detection: An Incremental Learning Framework,” IEEE Access, vol. 13, pp. 37872–37903, 2025, https://doi.org/10.1109/access.2025.3544221.
Wijethilaka, et al., 2025 - R. W. K. S. Wijethilaka, K. Yapa, and D. Siriwardena, “A cost effective machine learning based network intrusion detection system using Rasp berry Pi for real time analysis,” PLOS One, vol. 20, no. 12, p. e0331123, Dec. 2025, https://doi.org/10.1371/journal.pone.0331123.