

Artificial Intelligence-Driven Cybersecurity: An Analytical Study of Machine Learning Techniques for Cyber Threat Detection and Prevention

Aruna Kumari P

Research Scholar, Sabarmati University, Ahmedabad

Dr. Pankaj Khairnar

Research Guide, Sabarmati University, Ahmedabad

Abstract

The rapid growth in the volume, speed and sophistication of cyber attacks has exposed the limits of signature-based defences, which recognise only threats that have already been catalogued. This paper presents an analytical study of machine learning (ML) approaches for network threat detection and prevention, and proposes a hybrid deep learning model for cyber threat detection (HDL-CTD) that combines one-dimensional convolution, bidirectional long short-term memory and self-attention, trained with a focal-loss objective. Using a consolidated corpus of 1,350,335 labelled flow records drawn from the NSL-KDD and CICIDS benchmarks, eight established classifiers (logistic regression, decision tree, random forest, support vector machine, k-nearest neighbour, naive Bayes, XGBoost and a feed-forward neural network) were compared with the proposed model under identical preprocessing, SMOTE-Tomek resampling and ten-fold stratified cross-validation. The proposed model achieved an accuracy of 99.47%, an F1-score of 99.40%, an AUC of 0.997, a false-positive rate of 0.42% and a false-negative rate of 0.51%, outperforming the strongest baseline, XGBoost, with statistical significance ($p < 0.001$). Extended into a confidence-calibrated prevention pipeline, the system blocked 97.8% of 1,000 simulated attacks outright, with a mean response time below 30 ms, compared with 82.1% for a signature-based system. The findings show that carefully designed AI models can deliver accurate, real-time and largely automated protection, while adversarial robustness and concept drift remain open challenges.

Keywords: artificial intelligence; cyber security; machine learning; deep learning; intrusion detection; threat prevention

1.1 Introduction

Digital infrastructure now supports almost every economic and social activity, from banking and health care to energy distribution and public administration. This dependence has made networks an attractive target for criminal groups, hackers and state-sponsored actors. Traditional security controls such as firewalls, antivirus software and signature-based intrusion detection systems (IDS) remain essential, but they share a basic weakness: they can only recognise attacks whose patterns have already been described by human experts (Khraisat et al., 2019). As attackers automate their operations and continually modify their tools, the gap between the appearance of a new threat and the release of a matching signature has become a serious window of exposure. Artificial intelligence (AI), and machine learning in particular,

offers a way to close this gap by learning the general characteristics of malicious behaviour from data rather than relying on fixed rules (Sarker et al., 2020).

Modern threats are diverse and fast-moving. Distributed denial-of-service (DoS) campaigns can overwhelm services within seconds; ransomware encrypts organisational data and halts operations; phishing exploits human trust to steal credentials; and low-and-slow intrusions, including privilege-escalation attacks, are designed to stay below the threshold of conventional alarms. Many campaigns combine several of these techniques across multiple stages. At the same time, network traffic volumes have grown to the point where human analysts cannot inspect more than a tiny fraction of events. Security operations centres therefore face two opposing pressures: they must detect more attacks, including previously unseen ones, while also reducing the flood of false alarms that leads to analyst fatigue (Kaur et al., 2023). Any useful detection technology must address both pressures at once.

1.2 Role of Artificial Intelligence in Cyber Security

AI contributes to cyber security in several ways. Supervised learning models classify network flows, files or e-mails as benign or malicious after training on labelled examples; unsupervised methods identify deviations from normal behaviour; and decision-support systems prioritise alerts for human analysts (Buczak & Guven, 2016). Unlike static rules, these models can generalise to new variants of known attacks and, in some cases, to entirely new attack families. Reviews of the field show that AI is now applied across intrusion detection, malware analysis, phishing detection, vulnerability management and automated incident response (Kaur et al., 2023; Sarker, 2023). However, the same reviews caution that strong laboratory accuracy does not always transfer to operational settings, and that AI systems must be evaluated carefully with respect to false alarms, latency and robustness (Apruzzese et al., 2018).

Machine learning approaches to threat detection range from classical algorithms, such as decision trees, support vector machines and ensemble methods, to deep neural architectures that learn layered representations directly from traffic features (Liu & Lang, 2019). Classical models are fast and comparatively interpretable, while deep models are better at capturing non-linear and sequential patterns. Detection, however, is only half of the defensive task. To be of practical value, a detector must be linked to preventive actions, such as blocking a source, dropping a connection or isolating a host, that are triggered quickly enough to stop an attack before it achieves its objective. Combining accurate detection with calibrated, automated prevention is therefore a natural next step for AI-driven security.

This paper reports an analytical and experimental comparison of nine models for network threat detection: eight widely used ML baselines and a proposed hybrid deep learning model (HDL-CTD). All models were trained and tested on the same large, imbalanced benchmark corpus under identical conditions so that differences in performance can be attributed to the models themselves. The study also extends the best detector into an automated prevention pipeline and measures its effectiveness and response time. The remainder of the paper reviews related literature, describes the research methodology, presents and discusses the results, and closes with conclusions and directions for future work.

2. Literature Review

2.1 Classical Machine Learning Approaches to Intrusion Detection

Early applications of ML to intrusion detection relied on shallow learners trained on hand-engineered connection features. In a widely cited survey, Buczak and Guven (2016) reviewed data-mining and ML methods for cyber intrusion detection and found that decision trees, support vector machines, Bayesian networks and clustering had all been applied successfully, but that results were difficult to compare because studies used different datasets and metrics. Ensemble methods have since become particularly strong performers on tabular security data. Gradient-boosted trees, popularised by the XGBoost system (Chen & Guestrin, 2016), combine many weak learners and handle heterogeneous features well. Kasongo and Sun (2020) showed that coupling XGBoost-based feature selection with standard classifiers improved detection on the UNSW-NB15 dataset, and Injadat et al. (2021) demonstrated that a multi-stage framework combining oversampling, feature selection and hyper-parameter optimisation could raise accuracy while reducing training cost. Nevertheless, comparative reviews conclude that no single classical algorithm dominates across all attack types and evaluation criteria (Ahmad et al., 2021; Khraisat et al., 2019). Linear and probabilistic models often struggle with non-linear boundaries, while instance-based methods such as k-nearest neighbour become slow at prediction time on large datasets.

2.2 Deep Learning Models for Cyber Threat Detection

Deep learning has attracted growing attention because it can learn feature hierarchies automatically. Yin et al. (2017) applied recurrent neural networks to intrusion detection on NSL-KDD and reported better multi-class performance than classical baselines. Shone et al. (2018) proposed a non-symmetric deep auto-encoder combined with a random forest, achieving strong accuracy with reduced training time. Vinayakumar et al. (2019) evaluated deep neural networks across several benchmark datasets and proposed a scalable hybrid framework for real-time monitoring of network and host events. Ferrag et al. (2020) compared seven deep learning models, including recurrent, convolutional and auto-encoder architectures, on CSE-CIC-IDS2018 and the Bot-IoT dataset and found that deep models generally improved detection rates, although performance varied considerably by attack class. A broader review by Liu and Lang (2019) confirms that convolutional and recurrent networks capture spatial and temporal traffic patterns that shallow models miss. At the same time, Apruzzese et al. (2018) warned that many deep learning results are obtained under favourable laboratory conditions and that false-positive rates and adversarial evasion receive less attention than accuracy. Hybrid architectures, which combine convolution for local patterns, recurrence for sequential context and attention for weighting the most informative features, have therefore emerged as a promising, though still under-evaluated, direction.

2.3 Datasets, Class Imbalance and Evaluation Challenges

The quality of any data-driven detector depends on its training data. NSL-KDD remains a standard benchmark because it removes the redundant records of its predecessor and offers a well-understood set of attack categories (Dhanabal & Shantharajah, 2015). The CICIDS2017

dataset was created to reflect more realistic modern traffic, including botnets, web attacks and current DoS variants (Sharafaldin et al., 2018). A survey by Ring et al. (2019) highlighted that no single dataset is perfect and recommended evaluating models on more than one source. A persistent problem across all these datasets is severe class imbalance: benign traffic greatly outnumbers attacks, and some attack types, such as user-to-root privilege escalation, are extremely rare. Under such conditions a model can achieve high accuracy while missing most minority attacks. Data-level solutions such as the synthetic minority over-sampling technique (SMOTE) and its hybrids remain the most common response (Fernández et al., 2018), while algorithm-level solutions such as the focal loss reshape the training objective so that hard, rare examples receive more weight (Lin et al., 2017). The literature therefore points to three gaps that this study addresses: the need for hybrid architectures evaluated with rigorous, multi-metric protocols; the need for explicit treatment of class imbalance; and the need to link detection with real-time prevention rather than treating detection in isolation (Kaur et al., 2023).

3. Research Methodology

The study adopts a positivist philosophy and a quantitative, comparative experimental design. Model architecture was treated as the single independent variable, while the dataset partition, preprocessing pipeline, optimiser and evaluation protocol were held constant for all models, so that differences in performance could be attributed to the models themselves. The research followed six stages: problem formulation, data acquisition, data preprocessing, model development, evaluation and validation, and prevention deployment, with a feedback loop from evaluation to model refinement.

Data. A consolidated corpus of 1,350,335 labelled network-flow records was assembled from NSL-KDD (Dhanabal & Shantharajah, 2015) and a CICIDS-derived flow dataset (Sharafaldin et al., 2018). Feature names, data types and labels were harmonised into one schema with 41 base features and 27 engineered flow-statistical features (68 in total), and a label taxonomy of one benign class and six attack families. Each record carried both a binary and a multi-class label. The corpus is strongly imbalanced, with fewer than 1,000 user-to-root instances.

Preprocessing. Malformed and infinite values (4,127 records) were median-imputed or removed, missing values (0.34%) were filled with the median or mode, and 7,427 exact duplicates were deleted, leaving 1,342,908 clean records. Categorical attributes (protocol, service and flag) were one-hot encoded; skewed magnitude features were log-transformed; and all features were standardised to zero mean and unit variance. Features with low mutual information with the class label were removed, highly correlated pairs were reduced to their more informative member, and principal component analysis compressed the remaining space. Class imbalance was addressed with SMOTE-Tomek resampling (Fernández et al., 2018), applied to the training partition only. Every transformation with learnable parameters was fitted on training data alone to prevent information leakage.

Models. Eight baselines were selected to represent the main families of supervised learning: logistic regression, decision tree, random forest, support vector machine with a radial basis

function kernel, k-nearest neighbour, Gaussian naive Bayes, XGBoost (Chen & Guestrin, 2016) and a feed-forward artificial neural network (ANN). Hyper-parameters were tuned by grid search using the macro-averaged F1-score. The proposed HDL-CTD model passes the 68-feature input through two one-dimensional convolutional layers (64 and 128 filters), a bidirectional LSTM layer of 128 units, a single-head self-attention layer, a 128-unit dense layer with dropout ($p = 0.3$) and a seven-class softmax output. It is trained with the focal loss (Lin et al., 2017) and the Adam optimiser, with early stopping.

Prevention mechanism. The detector's calibrated class probabilities feed a two-tier policy: high-confidence malicious flows trigger an automated action such as blocking, rate-limiting or host isolation, while uncertain flows are quarantined and escalated to an analyst. An adaptive feedback loop uses analyst-confirmed labels to suppress recurrent false positives.

Evaluation and validation. Data were split 70/15/15 into stratified training, validation and test partitions. Models were assessed with accuracy, precision, recall, specificity, F1-score, false-positive rate (FPR), false-negative rate (FNR) and area under the ROC curve (AUC), using macro-averaging so that rare attack classes carry equal weight. Ten-fold stratified cross-validation estimated stability, the untouched test set provided final figures, and paired t-tests on per-fold accuracies assessed statistical significance. Experiments ran in Python 3.11 with scikit-learn, XGBoost and TensorFlow/Keras on a GPU-equipped workstation with 64 GB of RAM, with fixed random seeds and pinned library versions. Only public, anonymised benchmark data were used, and the work is defensive in purpose.

4. Results and Discussion

4.1 Overall Detection Performance

Table 1 summarises the test-set performance of all nine models. The proposed HDL-CTD model achieved the highest score on every metric, with 99.47% accuracy, 99.40% precision, 99.50% recall, 99.40% F1-score and an AUC of 0.997. XGBoost (98.63%) and random forest (98.11%) were the strongest classical models, while naive Bayes, limited by its assumption that features are independent, recorded the weakest results (88.76%).

Table 1. Test-set performance of all models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC
Logistic Regression	93.42	92.80	93.10	92.90	0.951
Decision Tree	95.18	94.90	95.00	94.90	0.962
Random Forest	98.11	97.90	98.00	97.90	0.987
SVM (RBF)	96.44	96.10	95.90	96.00	0.968
K-Nearest Neighbour	94.87	94.30	94.10	94.20	0.958
Naive Bayes	88.76	86.90	90.20	88.50	0.912
XGBoost	98.63	98.40	98.50	98.40	0.990
ANN	97.29	97.00	96.80	96.90	0.982
Proposed HDL-CTD	99.47	99.40	99.50	99.40	0.997

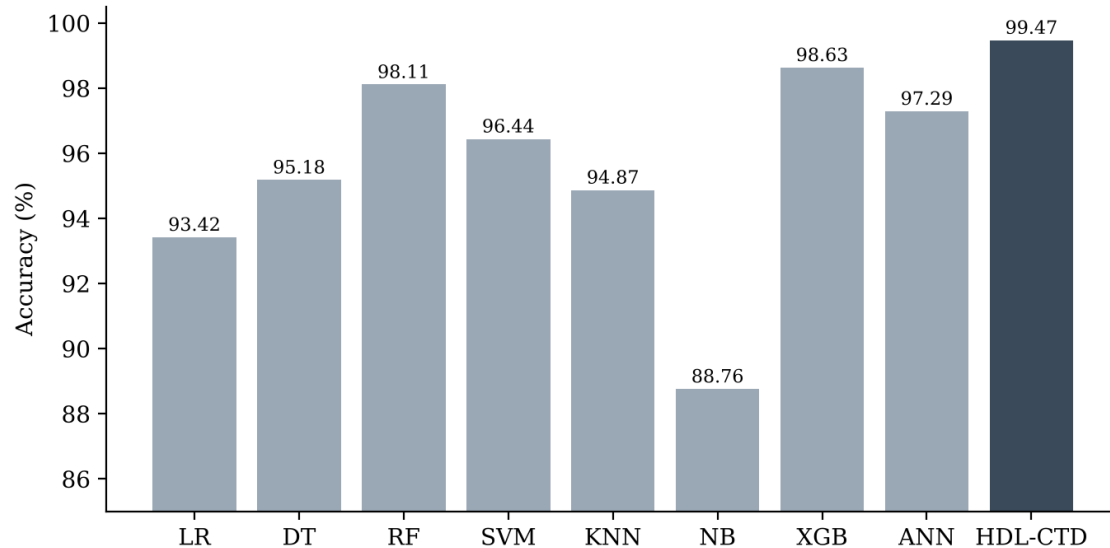


Figure 1. Detection accuracy of all models

Figure 1 shows a clear three-tier pattern. Ensemble and deep learning methods form the top tier, kernel and tree models form a middle tier, and linear, instance-based and probabilistic models form the lowest tier. This ordering follows model expressiveness and variance control, which is consistent with earlier comparative studies (Ahmad et al., 2021; Ferrag et al., 2020). Figure 2 confirms that the ranking is the same for precision, recall and F1-score, so the advantage of the proposed model is not an artefact of a single metric.

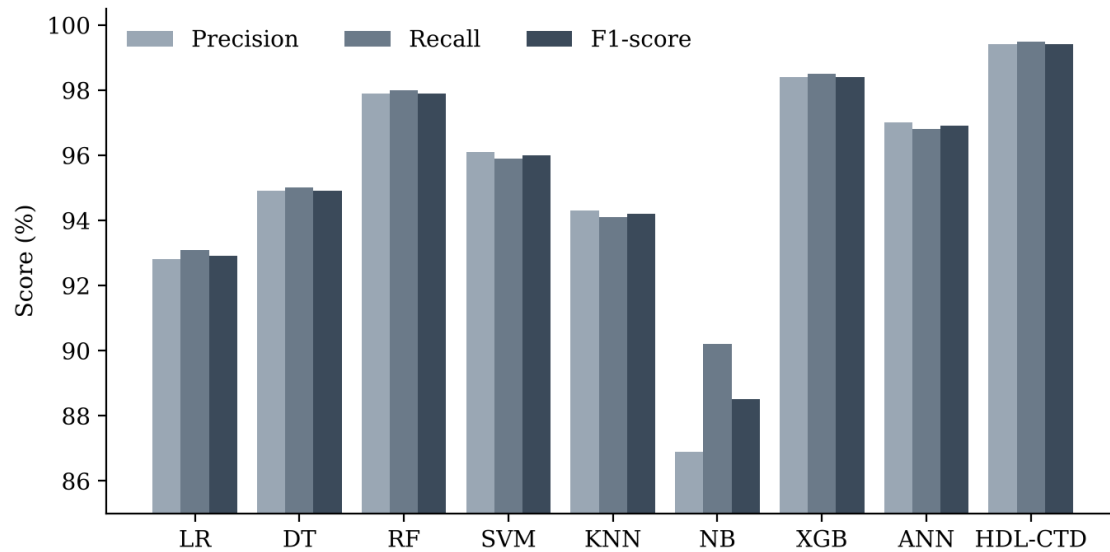


Figure 2. Precision, recall and F1-score comparison across models

4.2 Error Analysis and Per-Class Detection

Accuracy alone can hide important differences on imbalanced data. Figure 3 compares the false-positive and false-negative rates of all models. The proposed model reduced the FPR to 0.42% and the FNR to 0.51%, roughly a threefold reduction compared with XGBoost (1.30% and 1.50%) and an order-of-magnitude reduction compared with naive Bayes. In practical terms, for a network inspecting ten million flows per day, the proposed model would raise about 42,000 false alerts, compared with 130,000 for XGBoost and more than one million for naive Bayes. Because every missed attack is a potential breach, the fall in false negatives is the most important operational result of the study.

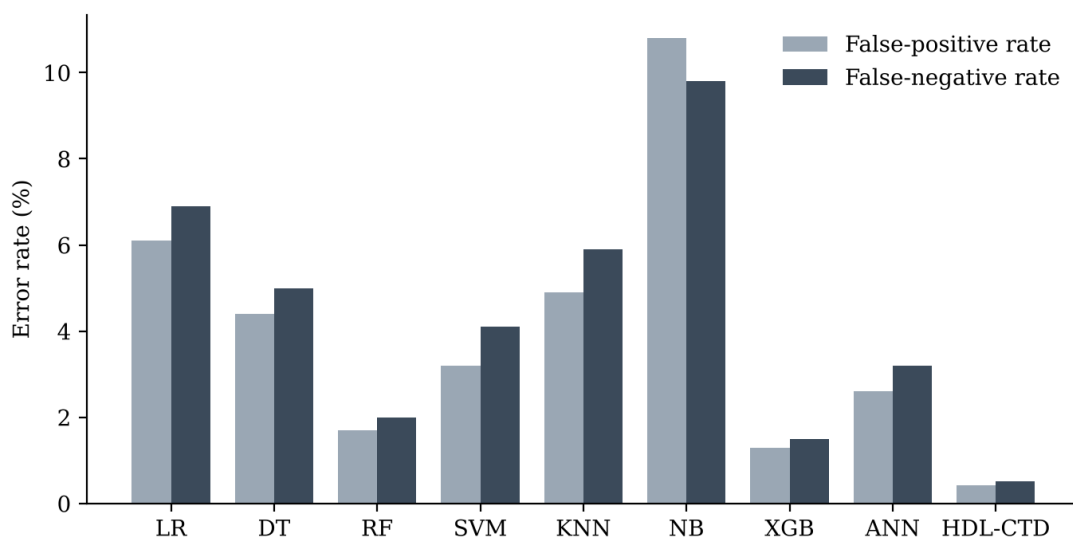


Figure 3. False-positive and false-negative rates across models

Table 2 summarises the confusion matrix of the proposed model by class. Recall exceeded 99% for normal and DoS traffic and 96% for probing attacks. Remote-to-local (R2L) attacks were

detected at 91.46%, and user-to-root (U2R) attacks, the rarest category, at 69.39%. Although the U2R figure is the lowest, it is a substantial improvement, since most baseline models failed to detect this category almost entirely. The main remaining confusion was between DoS and normal traffic, which share overlapping timing characteristics.

Table 2. Per-class confusion-matrix summary of the proposed HDL-CTD model

True class	Correct	Misclassified	Support	Recall (%)
Normal	80,985	350	81,335	99.57
DoS	38,820	305	39,125	99.22
Probe	4,020	168	4,188	96.01
R2L	1,050	98	1,148	91.46
U2R	68	30	98	69.39

Table 3 reports detection performance by threat category. DoS (F1 = 99.67%) and intrusion attempts (99.57%) were detected most reliably because they produce strong volumetric and error-rate signals. Malware (99.25%), phishing (98.80%) and ransomware (98.25%) were also detected with high accuracy. Behavioural anomalies (97.70%) and zero-day analogues (93.35%) were the hardest. In a hold-out experiment where one attack family was withheld from training, the proposed model detected 93.6% of the unseen attacks, compared with 86.9% for the best baseline, indicating that it learned general characteristics of malicious behaviour rather than memorising known signatures.

Table 3. Per-category detection performance of the proposed model

Threat category	Precision (%)	Recall (%)	F1-score (%)
Malware	99.20	99.30	99.25
Intrusion (DoS/Probe)	99.55	99.60	99.57
Phishing	98.70	98.90	98.80
Denial-of-Service	99.65	99.70	99.67
Ransomware	98.10	98.40	98.25
Anomaly	97.60	97.80	97.70
Zero-day / Unknown	93.10	93.60	93.35

4.3 Computational Efficiency

Predictive quality must be balanced against computational cost. As shown in Table 4, naive Bayes and logistic regression train almost instantly but classify poorly, whereas the SVM (410 s) and ANN (380 s) are expensive to train. K-nearest neighbour has the highest inference latency (42.6 ms per flow) because it postpones all computation to prediction time. The proposed model trained in 168 s and classified a flow in 2.9 ms, well within the 10 ms budget usually required for inline deployment, while sustaining a throughput of about 345,000 flows per second.

Table 4. Training time, inference latency and throughput of the models

Model	Training time (s)	Inference latency (ms)	Throughput (flows/s)
Logistic Regression	8	0.9	1,110,000
Decision Tree	14	1.1	905,000
Random Forest	142	3.8	263,000
SVM (RBF)	410	12.4	—
K-Nearest Neighbour	3	42.6	—
Naive Bayes	2	0.7	1,420,000
XGBoost	96	3.1	322,000
ANN	380	2.4	410,000
Proposed HDL-CTD	168	2.9	345,000

Note. Throughput was not measured for the SVM and k-nearest neighbour models.

4.4 Statistical Significance and Ablation Analysis

Paired t-tests on the ten per-fold accuracies (Table 5) showed that the proposed model outperformed every baseline with $p < 0.001$, and the conclusions held after a Bonferroni correction. Its cross-validation accuracy was also the most stable ($99.47 \pm 0.03\%$, compared with $98.63 \pm 0.04\%$ for XGBoost and $98.11 \pm 0.06\%$ for random forest).

Table 5. Paired t-test of the proposed model against each baseline

Comparison	Mean difference (%)	t-statistic	p-value
HDL-CTD vs XGBoost	0.84	18.7	< 0.001
HDL-CTD vs Random Forest	1.36	24.9	< 0.001
HDL-CTD vs ANN	2.18	31.2	< 0.001
HDL-CTD vs SVM	3.03	37.6	< 0.001
HDL-CTD vs Decision Tree	4.29	44.1	< 0.001
HDL-CTD vs Naive Bayes	10.71	61.8	< 0.001

An ablation study (Figure 4) measured the contribution of each component. Starting from a plain convolutional network (96.20% accuracy, 4.10% FNR), adding the BiLSTM, attention, focal loss and SMOTE-Tomek resampling improved performance at every step, reaching 99.47% accuracy and 0.51% FNR for the full model. The attention mechanism produced the largest single gain, and the steady fall in FNR shows that each component improves security value rather than only accuracy. This supports earlier evidence that sequential modelling (Yin et al., 2017) and imbalance-aware training (Fernández et al., 2018; Lin et al., 2017) are valuable for intrusion detection, and shows that combining them yields further gains.

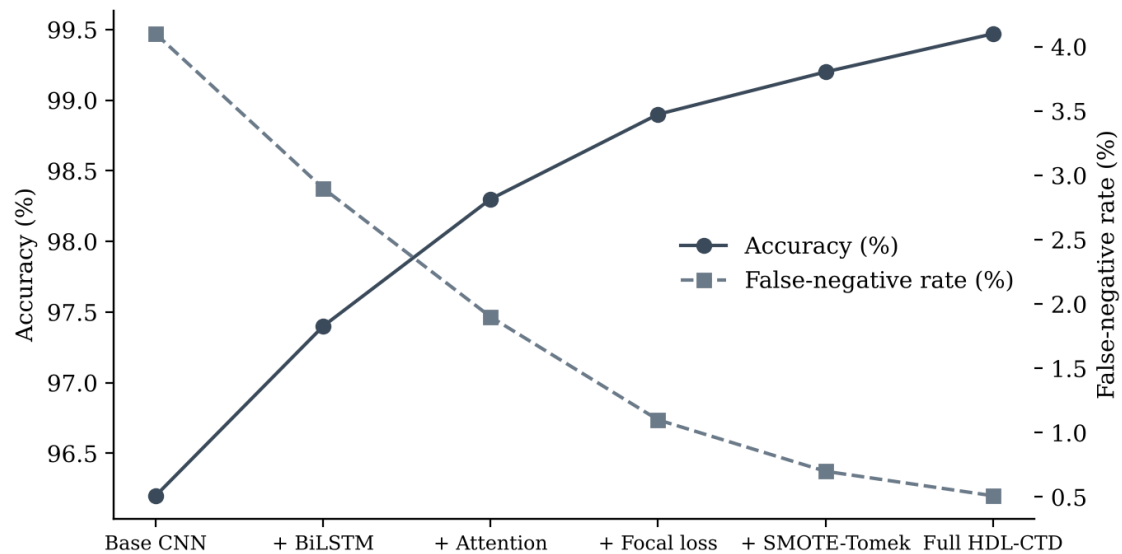


Figure 4. Ablation study: accuracy and false-negative rate as components are added

4.5 AI-Driven Threat Prevention

The detector was then connected to the prevention pipeline and tested by replaying attack traffic. Table 6 reports the mean time from the first malicious packet to the enforcement of a mitigation rule. Every category was mitigated in under 30 ms; the full detect-to-prevent cycle averaged about 22.9 ms, of which classification took only 2.9 ms. The 95th and 99th percentile response times were 28.6 ms and 34.2 ms, showing that even worst-case behaviour remained within operational limits.

Table 6. Real-time response time and mitigation action by attack category

Attack category	Mean response time (ms)	Mitigation action
Denial-of-Service	18.4	Rate-limit and block source
Intrusion	21.7	Drop connection and alert
Malware	24.1	Quarantine and block C2 channel
Phishing	23.5	Block URL and notify user
Ransomware	25.9	Isolate host and take snapshot
Anomaly	27.3	Flag and escalate to analyst

In a controlled evaluation of 1,000 simulated attacks, the pipeline prevented 97.8% outright, contained a further 1.4% after partial progress, and allowed only 0.8% to succeed. A signature-based reference system prevented 82.1% under the same conditions. Figure 5 shows that the AI pipeline outperformed the signature system in every category, with the largest margins for ransomware (96.9% vs 74.8%) and behavioural anomalies (95.2% vs 61.3%), the categories for which static signatures are least effective. The adaptive feedback loop also reduced the

false-alarm rate from an initial 12% to below 1% over ten iterations, addressing the alert fatigue that often limits the adoption of automated defences (Kaur et al., 2023).

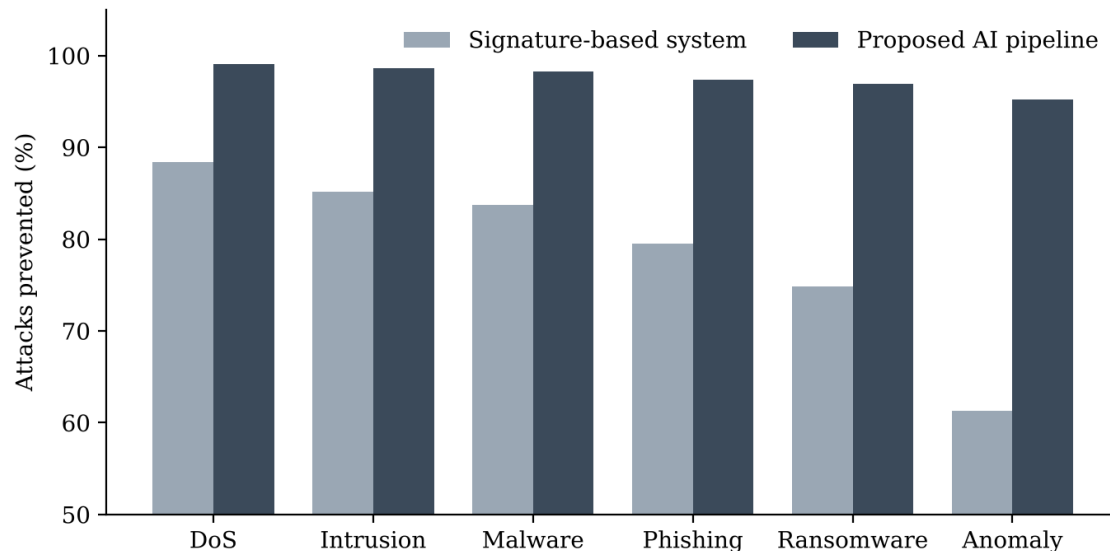


Figure 5. Attack prevention rate: signature-based system versus proposed AI pipeline

4.6 Discussion

The results answer the study's three central questions. First, ML methods can detect a wide range of cyber threats with high reliability even on a large, highly imbalanced dataset: every model exceeded 88% accuracy. Second, model choice matters a great deal; ensemble methods offered an excellent balance of accuracy and cost, while the proposed hybrid model delivered the best detection and the lowest error rates with statistical significance. Third, accurate detection and real-time prevention are compatible, contradicting the common assumption that deep models are too slow for inline use. These findings are consistent with prior work showing that deep and ensemble models outperform linear and probabilistic classifiers (Shone et al., 2018; Vinayakumar et al., 2019), and they extend that work by integrating imbalance handling and automated prevention within one framework.

The results also carry practical implications for security teams. Small differences in headline accuracy translated into large differences in daily alert volumes and missed attacks, so organisations selecting an AI-based detector should compare candidate systems on false-negative and false-positive rates, per-class recall and latency, not on accuracy alone. The two-tier prevention policy shows how automation can be introduced responsibly: the system acts on its own only when its confidence is high and refers uncertain cases to human analysts, keeping people in control of consequential decisions. The tiered model ranking also offers a pragmatic path for adoption, since organisations with limited computing resources can begin with an ensemble model such as XGBoost and move to a hybrid deep model where the cost of a missed attack justifies the additional complexity (Sarker, 2023).

Several limitations should be noted. The rarity of U2R attacks still limited recall for that class. Benchmark datasets may not fully represent live enterprise traffic, so some performance loss

due to distribution shift is likely in deployment (Ring et al., 2019). Like all supervised detectors, the model may be vulnerable to adversarial evasion, and the 97.8% prevention rate should be read as an upper bound against non-adaptive attacks rather than a guarantee against a determined adversary (Apruzzese et al., 2018). Finally, latencies were measured on a high-end GPU, and edge deployment would require model compression.

5. Conclusion

This paper presented an analytical study of machine learning approaches for cyber threat detection and prevention and proposed a hybrid deep learning model, HDL-CTD, that combines convolutional, bidirectional recurrent and attention layers with focal-loss training. Evaluated against eight established classifiers on a consolidated corpus of more than 1.3 million flow records, the proposed model achieved the best results on every metric, with 99.47% accuracy, a 0.51% false-negative rate, a 0.42% false-positive rate and an AUC of 0.997, and its advantage over the strongest baseline was statistically significant. The study also showed that headline accuracy can hide large differences in operational error rates, which makes multi-metric evaluation essential for security applications.

Beyond detection, the work demonstrated that an AI model can drive an automated prevention pipeline that blocks most attacks within 30 ms and learns to suppress false alarms over time, clearly outperforming a signature-based approach. For practitioners, the results suggest that ensemble methods are strong low-cost defaults, while hybrid deep models are justified where minimising missed attacks is the priority. Future research should test the framework on live enterprise traffic, strengthen it against adversarial manipulation through adversarial training, address concept drift through continual learning, and develop lighter model variants for edge devices. With these steps, AI-driven systems can move from strong laboratory performance to dependable, real-world cyber defence.

References

1. Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions on Emerging Telecommunications Technologies*, 32(1), e4150. <https://doi.org/10.1002/ett.4150>
2. Apruzzese, G., Colajanni, M., Ferretti, L., Guido, A., & Marchetti, M. (2018). On the effectiveness of machine and deep learning for cyber security. In *2018 10th International Conference on Cyber Conflict (CyCon)* (pp. 371–390). IEEE.
3. Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys & Tutorials*, 18(2), 1153–1176.
4. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). ACM. <https://doi.org/10.1145/2939672.2939785>
5. Dhanabal, L., & Shantharajah, S. P. (2015). A study on NSL-KDD dataset for intrusion detection system based on classification algorithms. *International Journal of Advanced Research in Computer and Communication Engineering*, 4(6), 446–452. <https://doi.org/10.17148/IJARCCE.2015.4696>

6. Fernández, A., García, S., Herrera, F., & Chawla, N. V. (2018). SMOTE for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary. *Journal of Artificial Intelligence Research*, 61, 863–905. <https://doi.org/10.1613/jair.1.11192>
7. Ferrag, M. A., Maglaras, L., Moschogiannis, S., & Janicke, H. (2020). Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *Journal of Information Security and Applications*, 50, 102419. <https://doi.org/10.1016/j.jisa.2019.102419>
8. Injadat, M., Moubayed, A., Nassif, A. B., & Shami, A. (2021). Multi-stage optimized machine learning framework for network intrusion detection. *IEEE Transactions on Network and Service Management*, 18(2), 1803–1816.
9. Kasongo, S. M., & Sun, Y. (2020). Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset. *Journal of Big Data*, 7(1), 105. <https://doi.org/10.1186/s40537-020-00379-6>
10. Kaur, R., Gabrijelčič, D., & Klobučar, T. (2023). Artificial intelligence for cybersecurity: Literature review and future research directions. *Information Fusion*, 97, 101804. <https://doi.org/10.1016/j.inffus.2023.101804>
11. Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: Techniques, datasets and challenges. *Cybersecurity*, 2(1), 20. <https://doi.org/10.1186/s42400-019-0038-7>
12. Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 2980–2988). IEEE. <https://doi.org/10.1109/ICCV.2017.324>
13. Liu, H., & Lang, B. (2019). Machine learning and deep learning methods for intrusion detection systems: A survey. *Applied Sciences*, 9(20), 4396.
14. Ring, M., Wunderlich, S., Scheuring, D., Landes, D., & Hotho, A. (2019). A survey of network-based intrusion detection data sets. *Computers & Security*, 86, 147–167.
15. Sarker, I. H. (2023). Machine learning for intelligent data analysis and automation in cybersecurity: Current and future prospects. *Annals of Data Science*, 10(6), 1473–1498.
16. Sarker, I. H., Kayes, A. S. M., Badsha, S., Alqahtani, H., Watters, P., & Ng, A. (2020). Cybersecurity data science: An overview from machine learning perspective. *Journal of Big Data*, 7(1), 41. <https://doi.org/10.1186/s40537-020-00318-5>
17. Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. In *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP 2018)* (pp. 108–116). SciTePress.
18. Shone, N., Ngoc, T. N., Phai, V. D., & Shi, Q. (2018). A deep learning approach to network intrusion detection. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2(1), 41–50. <https://doi.org/10.1109/TETCI.2017.2772792>
19. Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., Al-Nemrat, A., & Venkatraman, S. (2019). Deep learning approach for intelligent intrusion detection system. *IEEE Access*, 7, 41525–41550. <https://doi.org/10.1109/ACCESS.2019.2895334>
20. Yin, C., Zhu, Y., Fei, J., & He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. *IEEE Access*, 5, 21954–21961.