International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

AI-Based Detection of Known and Unknown Cyber Attacks: A Review of Anomaly Detection Deep Open-Set and Meta-Learning Methods Benchmark Datasets and Deployable Detection Systems

Ishika Kumbhare
UG Student
Dept. of Computer Science and Engineering (Data Science)
Guru Nanak Institute of Technology
Nagpur, India
kumbhareishika419@gmail.com

Om Kelzarkar
UG Student
Dept. of Computer Science and Engineering (Data Science)
Guru Nanak Institute of Technology
Nagpur, India
omkelzarkar246@gmail.com

Pranav Khandare
UG Student
Dept. of Computer Science and Engineering (Data Science)
Guru Nanak Institute of Technology
Nagpur, India
pranavkhandare61@gmail.com

Prof. Ruchira Dabhade
Project Guide
Dept. of Computer Science and Engineering (Data Science)
Guru Nanak Institute of Technology
Nagpur, India

***Abstract*—**Cyber attacks are becoming more automated and adaptive and signature-based security tools can miss attacks whose patterns have not been seen before. This review surveys research on AI-based detection of known and unknown cyber attacks. We organise the literature into eight themes: the threat landscape of unknown and zero-day attacks supervised and ensemble learning unsupervised and one-class anomaly detection, deep representation learning open-set recognition and metric learning few-shot meta and reinforcement learning benchmark datasets and evaluation and deployment explainability and governance. For each theme we summarise representative work from twenty recent studies (2024–2026) the techniques used and the limitations reported by the original authors. We then compare detection approaches and datasets and use a proposed web-based system built with Python Scikit-learn and Flask as a reference case that shows how the themes combine in one pipeline. The review identifies open challenges including generalization across datasets evaluation on truly unseen attacks false alarms concept drift adversarial robustness explainability and lightweight real-time deployment. We conclude with research directions for trustworthy and deployable detection of unknown cyber attacks.

***Keywords*—**Cyber attack detection, intrusion detection zero-day attacks unknown attack detection anomaly detection machine learning deep learning open-set recognition meta-learning survey.

I. Introduction

Cyber security protects computers networks applications and data from unauthorised access modification and disruption. Modern organisations generate large volumes of network traffic and system logs and monitoring them manually is slow and error-prone. Traditional intrusion detection systems (IDS) compare observed activity with a database of known attack signatures or rules [1]. This works well when an attack has been seen before but it leaves a detection gap when an attack is new or modified. Zero-day attacks which exploit vulnerabilities not yet known to the software vendor are the most serious case and an empirical study of real-world attacks found that such exploits can stay undetected for long periods before they are disclosed [2].

Machine learning (ML) offers a behaviour-based alternative. Instead of matching signatures a model learns patterns of normal or malicious behaviour from data and flags traffic that departs from them [3] [4]. Recent systematic reviews report that AI methods are promising for zero-day detection but are limited by data availability algorithmic complexity and real-time use [5] and the wider literature on machine learning for security is now very large [6]. At the same time anomaly detection in networks is harder than in many other fields because normal traffic is diverse and every false alarm costs analysts time [7].

Research on unknown-attack detection is extensive but spread across anomaly detection deep learning open-set recognition few-shot learning reinforcement learning and IoT security. Earlier reviews describe the methods in general [5] [8] but there is little consolidated guidance that links the limitations reported in individual studies to the design of a small, deployable detection system. This paper reviews twenty recent studies published between 2024 and 2026 with three objectives: (i) to organise the work into a taxonomy of detection approaches; (ii) to compare approaches and datasets and their trade-offs; and (iii) to identify open problems. We use a proposed web-based system for detecting known and unknown cyber attacks as a running reference case.

A. Review Methodology

The review follows the planning conducting and reporting stages recommended for structured literature reviews [9]. The core of the corpus is twenty peer-reviewed studies on unknown zero-day or anomaly-based intrusion detection published in venues such as IEEE Access Scientific Reports Computers & Security Computer Networks and PLOS ONE. This core is supplemented by foundational method papers dataset papers recent surveys and guidance from NIST. Inclusion criteria were relevance to at least one theme in the taxonomy reported evaluation results and a verifiable published source. Performance values and limitations quoted for each study are those stated by the original authors. They were produced on different datasets data splits and definitions of an unknown attack so they are not directly comparable. The review is therefore integrative rather than a quantitative meta-analysis.

B. Contributions

1) An eight-theme taxonomy of AI-based detection of known and unknown cyber attacks (Fig. 1).
2) A thematic synthesis of twenty recent studies with the results and limitations reported by the authors (Table I).
3) Comparative analyses of detection approaches (Table II) and benchmark datasets (Table III).
4) A mapping of the themes to a proposed reference system (Fig. 2 Table IV) its initial result and a set of open challenges and research directions.

II. Background

A. Known Unknown and Zero-Day Attacks

A known attack is one whose signature rule or behaviour is already represented in the detection system. An unknown attack is one whose behaviour is not sufficiently represented in the

training data or rules. The term covers zero-day exploits modified variants of known malware and new behaviours that have not been seen in a given network [5]. A signature-based IDS looks for predefined indicators while an anomaly-based IDS models legitimate behaviour and reports significant deviations [1] [3]. Because an anomaly detector does not need the name of the attack it can raise an alert for an attack that was never labelled in training. However an anomaly is not always an attack since rare legitimate activity can also be flagged.

B. Classification Anomaly Detection and Open-Set Recognition

Detection is usually posed in one of three ways. Closed-set classification assigns each record to one of the attack classes seen in training, so a completely new attack is forced into an existing class or treated as normal. Anomaly detection learns a profile of normal behaviour and scores how far a record departs from it [4]. Open-set recognition extends a classifier with an explicit unknown outcome [10], [11]. The input is typically a set of flow-level features, such as the duration of a connection, the bytes sent by the source, the bytes returned by the destination, and the number of connections in a time window.

C. Evaluation and Generalization

Detectors are commonly reported with accuracy, precision, recall, F1-score, area under the ROC curve (AUC), and false-positive rate. Studies on unknown attacks add a zero-day detection rate and inference latency. Accuracy alone can mislead when attacks are rare, and a random train–test split does not reproduce the zero-day situation, because every attack family appears in both parts. Stronger protocols withhold attack families, use time-based splits, and test across datasets [12], [5]. These protocols are needed to judge whether a detector will work on traffic it has not seen.

III. Taxonomy of Detection Approaches

Fig. 1 presents the taxonomy used to structure this review. The eight themes cover the nature of the problem (the threat landscape and unknown attacks), the main learning paradigms used for detection (supervised and ensemble learning, unsupervised and one-class detection, deep representation learning, open-set and metric learning, and few-shot, meta and reinforcement learning), how detectors are evaluated (datasets and evaluation), and how they are put to use (deployment, explainability, and governance).

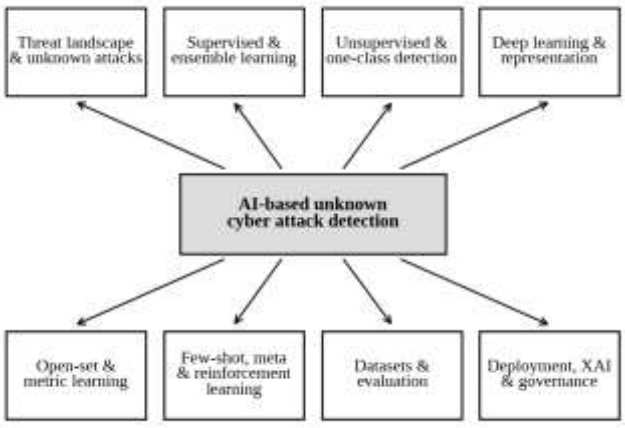


Fig. 1. Taxonomy of AI-based known and unknown cyber attack detection.

IV. Thematic Review

A. Threat Landscape and Unknown Attacks

Unknown attacks are difficult because defenders have no earlier sample to learn from. Bilge and Dumitraş [2] studied zero-day attacks in the wild and showed that exploits can circulate for a long time before the vulnerability is disclosed, a period in which signatures are usually unavailable. Por *et al.* [5] reviewed AI-based zero-day detection and found that machine learning is promising but limited by data availability, algorithmic complexity, and real-time application. Al Siam *et al.* [8] synthesised convolutional, deep, Bayesian, and reinforcement-learning models under a systematic protocol with time-split and cross-dataset tests. They report that deep neural networks reach nearly 99.56% accuracy on CICDDoS2019 and that reinforcement learning is promising for adaptive detection and response when safe training environments exist. Sommer and Paxson [7] warn that the cost of errors and the diversity of normal traffic make network anomaly detection harder than it first appears, a point that recurs in the studies reviewed below.

B. Supervised and Ensemble Learning

Supervised learners such as decision trees [13] and random forests [14] learn from labelled examples and are fast and fairly interpretable, but a closed-set model has no class for an attack it has never seen. Dai *et al.* [15] address this by combining an autoencoder with Random Forest and XGBoost classifiers. On unseen data they report 99.9892% accuracy, 100% precision, 99.9803% recall, and a 99.9901% F1-score, and they note that more diverse datasets and real-world validation are required. Touré *et al.* [16] combine supervised, unsupervised, and online learning on network flows and report 98.4% online-learning accuracy on an IBM dataset and 96.6% on NSL-KDD, with an average of 71 anomalous flows per target. Because the IBM data is confidential, the authors call for more public and diverse datasets. Ban *et al.* [17] combine adaptive particle swarm optimisation, a convolutional neural network (CNN), and squeeze-and-excitation blocks for IoT intrusion detection and report 88.73% accuracy and an 88.53% F1-score, but state that explicit zero-day and cross-dataset evaluation is still needed.

C. Unsupervised and One-Class Anomaly Detection

Unsupervised and one-class methods learn what normal traffic looks like and flag deviations, so they need few or no attack labels. Isolation Forest [18] isolates anomalies through random partitioning, and the one-class support vector machine (OC-SVM) [19] learns a boundary around normal data. Roopak *et al.* [20] combine random projection with K-Means, a Gaussian mixture model, and an OC-SVM ensemble for zero-day DDoS attacks in IoT networks. On CIC-DDoS2019 they report 94.55% accuracy, a 94.30% F1-score, and an ROC score of 0.945, but the study is limited to DDoS, so other attack types remain untested. Uddin *et al.* [21] propose usfAD, a one-class detector trained mainly on benign traffic, and report unknown-attack accuracy of 95.92% on NSL-KDD, 82.15% on UNSW-NB15, 98.69% on CIC-DDoS2019, and 99.43% on ToN-IoT. This spread of about 17 percentage points is consistent with the authors' note that performance varies by dataset. Tripathy *et al.* [22] extract features with a CNN and classify with an OC-SVM, reporting 98.41% accuracy on NSL-KDD and 99.31% on CIC-IDS2017, while noting that real-time detection and changing attack environments still need validation. Li *et al.* [23] train a GAN and Transformer model mainly on normal traffic and report F1-scores of 95.2% on CIC-IDS2017 and 83.2% on Kitsune. In a real-vehicle deployment the false-positive rate

was 13%, and the authors identify false alarms and cross-dataset generalization as the main weaknesses.

D. Deep Representation Learning

Autoencoders compress their input and reconstruct it. When they are trained mainly on normal traffic, they reconstruct normal records well and unusual records poorly, so the reconstruction error acts as an anomaly score. Kitsune used an ensemble of small autoencoders for online detection [24], and Hindy *et al.* [25] report zero-day detection accuracy of 89–99% on NSL-KDD and 75–98% on CICIDS2017 with an autoencoder. Yang *et al.* [26] combine a variational autoencoder [27] with attention fusion [28] and report 97.83% accuracy, a 96.42% F1-score, and an AUC of 0.9871 on CICIDS2017. With only 10% of the training data the F1-score is still 90.23%, which suggests good data efficiency, although more diverse environments and production-scale validation are needed. Almuflih *et al.* [29] use binary snake optimisation for feature selection and an attention-based bidirectional gated recurrent unit (Attention-BiGRU) for IoT zero-day detection. On ToN-IoT they report 98.28% accuracy, 91.39% precision, 91.39% recall, a 91.38% F1-score, and 95.22% AUC, with real-time deployment, scalability, and dynamic IoT validation left open. Krishnan *et al.* [30] add explainable AI to an attention fusion model and report 99.9966% accuracy for binary classification but only 71.0926% accuracy and a 74.4112% F1-score for zero-day multiclass classification. This large gap shows that deciding that traffic is malicious is much easier than deciding what kind of attack it is.

E. Open-Set Recognition and Metric Learning

Open-set methods add an unknown decision to a classifier [10]. OpenMax [11] does this for deep networks by estimating the probability that an input belongs to none of the known classes. Fang and Xie [31] combine InfoGAN with OpenMax for unknown intrusion traffic and report unknown-class accuracy of about 79–91%, depending on the attack class and model. They note that normal, known, and unknown traffic can still be confused. Alsaleh [32] uses a CNN with SMOTE oversampling [33], maximum mean discrepancy (MMD), and nearest-class-mean classification, and reports about 94% average accuracy on AWID with the detection rate reaching the reported full rate. Reducing false negatives and broader real-world validation are listed as needs. Alshammari and Alsaleh [34] apply a CNN, SMOTE, and open-set classification to the Internet of Vehicles (IoV) and report 94% accuracy, 100% detection, and a 3% false alarm rate on AWID, and 91.7% accuracy on NSL-KDD. Because the study focuses on IoV, other environments need testing. Zhang *et al.* [35] learn a distance-based embedding with a one-dimensional CNN, an improved triplet loss, and Mahalanobis distance. They reach average accuracy of 97.28% on CICIDS2017 but 85.47% on UNSW-NB15 and call for stronger cross-dataset generalization.

F. Few-Shot Meta and Reinforcement Learning

Few-shot and meta-learning methods aim to recognise a new attack from only a few examples. Model-agnostic meta-learning (MAML) [36] learns an initialisation that adapts quickly, prototypical networks [37] classify by distance to class prototypes, and Siamese networks [38] compare pairs of samples. Yan *et al.* [39] build a meta-learning model with sample-generation, feature-mapping, and feature-metric modules for the 5G-enabled industrial internet and report accuracy of up to 92.86% for the evaluated DoS category. More attack categories and real industrial-network tests are needed. Ijaz *et al.* [40] compare MAML, prototypical networks, and a Siamese network under data scarcity. MAML scores 96.67% accuracy and 97.54% recall on CICIDS2017, 92.87% and 92.99% on CICIoT2023, and 83.20% and 83.50% on CIC-UNSW-NB15, a fall of about 13 points across environments. Cevallos *et al.* [41] propose NERO, a hybrid of neural algorithmic reasoning, graph learning, and meta-learning for IoT zero-day detection. It is evaluated in two zero-day scenarios (Type-A and Type-B) with balanced accuracy, and the authors call for more diverse IoT traffic and real zero-day scenarios.

Reinforcement learning (RL) treats detection as a sequence of decisions that improve through feedback. Alam *et al.* [42] use deep RL for zero-day detection in network IDS and report accuracy and F1-score of about 0.95–0.99 across NetFlow-based datasets, but note that computational cost may limit real-time use. Hossain *et al.* [43] combine Siamese anomaly correlation with a proximal policy optimisation (PPO) agent. They report 99.07% accuracy on unseen attacks, a 93.94% zero-day detection rate, a 2.21% false-positive rate, and 0.50 ms latency, and they call for more real-world and large-scale validation.

G. Datasets and Evaluation

Dataset choice shapes what a detector learns. NSL-KDD [44] is a cleaned version of the 1999 KDD Cup data and is still widely used despite its age. UNSW-NB15 [45] adds modern attack families with engineered features, while CIC-IDS2017 [46], CIC-DDoS2019 [47], AWID [48], ToN-IoT [49], and CICIoT2023 [50] cover enterprise, DDoS, wireless, and IoT settings. Table III summarises these resources. Among the twenty studies in Table I, CIC-IDS2017 is used in five and NSL-KDD in four, so results are concentrated on a small set of benchmarks. Engelen *et al.* [51] revisited CIC-IDS2017 and found problems in traffic generation, flow construction, feature extraction, and labelling that distort model evaluation. More generally, Arp *et al.* [12] list recurring pitfalls in security machine learning, including sampling bias, label inaccuracy, data snooping, the base-rate fallacy, and evaluation only in the laboratory. Standardised NetFlow feature sets [52] make it easier to train on one dataset and test on another.

H. Deployment Explainability and Governance

A detector is useful only if analysts can use it. Practical systems wrap a trained model in an API and an interface, return a clear label or risk score with supporting evidence, and keep logs for review. Ohtani *et al.* [53] propose IDAC, which extracts attack candidates from each IoT network and shares them through federated learning [54] so that networks with different device types can benefit from each other's detections. They report F1-scores above 81% for attack types and a false-positive rate below 9%, and they identify communication overhead and robustness to unknown attacks as open questions. Explainability helps analysts trust and investigate alerts. Krishnan *et al.* [30] pair attention fusion with explainable AI, and feature-attribution methods such as SHAP [55] can show which flow features drove a decision. Normal behaviour also changes over time, so deployed models must be monitored and updated to handle concept drift [56]. Alerts can be mapped to adversary tactics and techniques with the MITRE ATT&CK knowledge base [57] to give analysts context. The NIST AI Risk Management Framework [58] lists characteristics of trustworthy AI, including validity and reliability, security and resilience, explainability, and privacy, which apply directly to detection tools that handle sensitive network data. Finally, latency matters: Hossain *et al.* [43] report 0.50 ms per decision, whereas Alam *et al.* [42] flag computational cost as a barrier.

TABLE I
SUMMARY OF REVIEWED STUDIES, REPORTED RESULTS, AND LIMITATIONS

Study	Technique and reported result	Reported limitation
Dai *et al.* [15]	Autoencoder + RF/XGBoost; 99.99% F1 on unseen data	Needs diverse data and real-world validation
Touré *et al.* [16]	Supervised, unsupervised and online learning; 98.4% (IBM), 96.6% (NSL-KDD)	Confidential IBM data; needs public datasets
Ban *et al.* [17]	APSO-CNN-SE; 88.73% accuracy	No explicit zero-day or cross-dataset test
Roopak *et al.* [20]	Random projection + K-Means, GMM, OC-SVM; 94.55% (CIC-DDoS2019)	Zero-day DDoS only
Uddin *et al.* [21]	usfAD one-class model; 82.15–99.43% accuracy over four datasets	Varies by dataset; needs large-scale validation
Tripathy *et al.* [22]	CNN features + OC-SVM; 99.31% (CIC-IDS2017)	Real-time and changing environments not validated
Li *et al.* [23]	GAN + Transformer on normal traffic; F1 95.2% (CIC-IDS2017)	13% false-positive rate on a real vehicle; cross-dataset drop
Yang *et al.* [26]	VAE + attention fusion; F1 96.42% (CICIDS2017)	Needs diverse, production-scale tests
Almuflih *et al.* [29]	Snake optimisation + Attention-BiGRU; 98.28% (ToN-IoT)	Real-time use, scalability, dynamic IoT
Krishnan *et al.* [30]	Attention fusion + XAI; 99.9966% binary, 71.09% zero-day multiclass	Large binary vs. multiclass gap
Fang and Xie [31]	InfoGAN + OpenMax; about 79–91% unknown-class accuracy	Normal, known and unknown traffic confused
Alsaleh [32]	CNN, SMOTE, MMD, nearest class mean; about 94% (AWID)	False negatives; real-world validation
Alshammari and Alsaleh [34]	CNN, SMOTE, open-set; 94% accuracy, 3% false alarms (AWID)	Focused on IoV only
Zhang *et al.* [35]	1D-CNN, triplet loss, Mahalanobis; 97.28% (CICIDS2017), 85.47% (UNSW-NB15)	Dataset-dependent performance
Yan *et al.* [39]	Meta-learning; up to 92.86% (DoS category)	More attack types; real industrial tests
Ijaz *et al.* [40]	MAML, prototypical, Siamese; MAML 96.67% (CICIDS2017)	Cross-environment generalization with little data
Cevallos *et al.* [41]	NERO: neural algorithmic reasoning, graph and meta-learning	More diverse IoT traffic; real zero-days
Alam *et al.* [42]	Deep RL; accuracy and F1 about 0.95–0.99 (NetFlow sets)	Computational cost for real-time use
Hossain *et al.* [43]	Siamese + PPO RL; 99.07% accuracy, 0.50 ms latency	More real-world, large-scale validation
Ohtani *et al.* [53]	Federated learning + anomaly extraction; F1 above 81%, FPR below 9%	Communication overhead; unknown-attack robustness

V. Comparative Analysis

Table II compares the main families of detection approaches discussed in Section IV, and Table III lists the benchmark datasets. The comparison is qualitative and reflects the results and limitations reported by the original authors rather than a new experiment.

TABLE II
COMPARISON OF DETECTION APPROACHES

Approach	Strengths	Weaknesses
Supervised / ensemble [15], [16]	Accurate and fast on known classes; interpretable trees	Closed-set; needs labels for every attack class
Unsupervised / one-class [21], [22]	Needs mainly normal traffic; flags novel behaviour	Threshold-sensitive; false alarms; varies by dataset
Autoencoder / VAE / attention [26], [30]	Learns complex normal patterns; strong benchmark scores	Needs data and compute; weak at naming unseen attacks
Open-set / metric learning [31], [35]	Explicit unknown class; distance-based decisions	Confuses normal, known and unknown; dataset-tuned
Few-shot / meta-learning [40], [39]	Adapts to new attacks from few samples	Accuracy falls across environments; training overhead
Reinforcement learning [42], [43]	Adaptive policy; low latency reported	Computational cost; reward design; safe training
Federated learning [53]	Shares knowledge across networks	Communication overhead; unknown-attack robustness untested

TABLE III
BENCHMARK DATASETS FOR UNKNOWN ATTACK DETECTION

Dataset	Type	Notes
NSL-KDD [44]	Network records	Cleaned KDD Cup 99; dated attack types
UNSW-NB15 [45]	Network flows	Nine attack families; engineered features
CIC-IDS2017 [46]	Network flows	Benign and multiple attacks; labelling and flow errors reported [51]
CIC-DDoS2019 [47]	Network flows	Realistic DDoS attacks and a DDoS taxonomy
AWID [48]	Wi-Fi (802.11)	Real wireless-network attacks
ToN-IoT [49]	IoT / IIoT	Telemetry, operating-system and network data
CICIoT2023 [50]	IoT network	Real-time IoT attack benchmark; large scale
Kitsune [24]	IoT packet traces	Online anomaly-detection benchmark
NetFlow (NF) sets [52]	NetFlow records	Common feature set for cross-dataset tests

Three patterns emerge from Tables I–III. First, the highest scores come from binary or in-distribution tests: Dai *et al.* [15] and Krishnan *et al.* [30] both report above 99.9% on one evaluation set. Second, scores fall when the task changes from detecting an attack to naming an unseen one, or when the data source changes: 71.09% for zero-day multiclass classification [30], 85.47% on UNSW-NB15 against 97.28% on CICIDS2017 [35], and a fall of about 13 points across environments for MAML [40]. Third, evidence for deployment is thin. Real-deployment results and latency measurements are reported by few of the studies, for example the real-vehicle test of Li *et al.* [23] and the 0.50 ms latency of Hossain *et al.* [43]. For a practical system, these patterns favour a layered design in which a supervised model handles known attacks, an anomaly model handles unknown behaviour, and evaluation uses held-out attack families and cross-dataset tests.

VI. Reference Case: Proposed Detection System

The reference case is a proposed web-based system, AI-based detection of known and unknown cyber attacks, that analyses network-traffic features and reports whether a connection is normal traffic, a known cyber attack, or a potential unknown attack. Its method follows the stages of problem identification, data collection, data pre-processing, model training, model testing, attack detection (known or unknown), and result visualisation. Fig. 2 shows the processing pipeline, and Table IV maps its modules to the themes of this review.

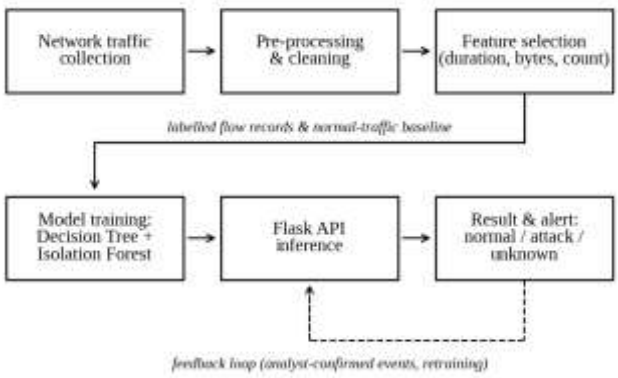


Fig. 2. Processing pipeline of the proposed detection system.

TABLE IV
MAPPING OF REVIEW THEMES TO THE PROPOSED SYSTEM

Theme	System module	Key technique
Threat landscape	Problem definition	Flow behaviour instead of signatures
Supervised learning	Known-attack classifier	Decision Tree (Scikit-learn)
Unsupervised detection	Unknown-attack detector	Isolation Forest with a threshold
Deep learning	Planned extension	Autoencoder / VAE comparison
Open-set output	Three-way result	Normal / attack / potential unknown
Meta and RL	Planned adaptation	Retraining on analyst-confirmed events
Datasets	Data ingestion	Pandas, NumPy; CSV; planned UNSW-NB15 and CIC-IDS2017
Deployment, governance	API, interface, alerts	Flask, HTML/CSS/JS, Joblib; user roles; logs

The prototype is written in Python. Pandas and NumPy handle data processing, Scikit-learn [59] provides the models, Joblib stores the trained models, and Flask connects a web interface built with HTML, CSS, and JavaScript to the models through a prediction API. The user enters four flow features, namely duration, source bytes, destination bytes, and connection count, and presses a Detect Attack button. The interface returns the result as Normal Traffic, Cyber Attack, or Potential Unknown Attack. A Decision Tree with a maximum depth of five, trained on an 80/20 train–test split, is used for known attacks, and an Isolation Forest [18] with a contamination setting of 0.1 is used to flag behaviour that departs from the training data. Student and administrator roles, a users table, and a table of detection records support login, monitoring, and review, and unit tests cover input validation, the prediction module, and the Flask API.

The only measured result so far is an initial one: on a small sample dataset the Decision Tree reached about 66.67% accuracy through the Flask interface. This figure is not a general performance claim, because the dataset size, confusion matrix, precision, recall, and F1-score have not yet been recorded. The intended outcomes are near real-time monitoring of network traffic, early alerts, fewer false alarms, and better protection against emerging threats, and these must be established by controlled evaluation. This paper reports no other measured results for the system.

Two observations follow from the case. First, the two models answer different questions. The Decision Tree answers which known class a record belongs to, while the Isolation Forest answers whether the record is unlike the training data. A rule for combining them is needed, for example reporting a record that the Decision Tree labels normal but the Isolation Forest flags as an outlier as a potential unknown attack. The Isolation Forest should preferably be fitted on normal traffic only and its threshold calibrated on validation data, as discussed in Section IV-C, because fitting it on mixed data with a fixed contamination setting assumes that about 10% of traffic is anomalous. Second, a detector that scores well on its training data can still fail on unseen attacks. The system should therefore be tested on modern datasets such as UNSW-NB15 [45] and CIC-IDS2017 [46], with the labelling corrections discussed in [51], across datasets, and on attack families withheld from training [12]. Until then, the 66.67% figure is best read as a starting point.

VII. Open Challenges and Research Directions

- **Generalization across datasets.** Reported accuracy for the same method differs by about 13 to 17 points between datasets and environments, so detectors need validation across datasets and networks [21], [35], [40].
- **Evaluation on truly unseen attacks.** Protocols that withhold attack families, use time-based splits, and report a zero-day detection rate are needed in place of random splits [12], [5], [8].
- **False alarms and alert fatigue.** Threshold calibration and cost-aware metrics are needed; a 13% false-positive rate in a real deployment shows the scale of the problem [23], [7].
- **Naming unseen attacks.** The gap between binary detection and multiclass identification of zero-day attacks must be closed [30], [31].
- **Concept drift and adversarial robustness.** Models must adapt as normal behaviour changes, without letting attackers poison the update data or evade the detector [56], [60].
- **Explainable, trustworthy alerts.** Feature-level explanations, calibrated risk scores, ATT&CK mapping, and human review of flagged events should accompany each alert [30], [55], [57], [58].
- **Lightweight real-time and federated deployment.** Efficient models and privacy-preserving collaboration are needed for edge and IoT settings [42], [43], [53], [29].
- **Empirical evaluation of the integrated system.** Controlled studies of the proposed system should report confusion matrices, precision, recall, false-positive cost, latency, and cross-dataset results.

VIII. Conclusion

Detecting known and unknown cyber attacks sits at the intersection of machine learning, network security, and data engineering. This review organised twenty recent studies and supporting literature into eight themes, compared detection approaches and datasets, and illustrated their integration through a proposed web-based detection system. The analysis shows that supervised, one-class, deep, open-set, meta-learning, and reinforcement-learning methods each solve part of the problem, and that reported scores are highest in binary or in-distribution tests. Generalization across datasets, identification of truly unseen attacks, false-alarm control, concept drift, adversarial robustness, and lightweight real-time deployment remain unresolved. For the reference system, the next steps are to evaluate it on modern datasets with held-out attack families, to report precision, recall, F1-score, false-positive rate, and latency, to add explainable outputs, and to test continuous learning under controlled conditions. No single model can be expected to detect every future attack, but a layered, explainable, and continuously evaluated system can

raise the chance that abnormal behaviour is noticed early and give analysts useful evidence.

REFERENCES

[1] K. Scarfone and P. Mell, "Guide to intrusion detection and prevention systems (IDPS)," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Spec. Publ. 800-94, Feb. 2007.

[2] L. Bilge and T. Dumitraş, "Before we knew it: An empirical study of zero-day attacks in the real world," in *Proc. ACM Conf. Comput. Commun. Secur. (CCS)*, 2012, pp. 833–844.

[3] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1153–1176, 2016.

[4] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, Art. no. 15, 2009.

[5] L. Y. Por, Z. Dai, S. J. Leem, Y. Chen, J. Yang, F. Binbeshr, K. Y. Phan, and C. S. Ku, "A systematic literature review on AI-based methods and challenges in detecting zero-day attacks," *IEEE Access*, vol. 12, pp. 144150–144163, 2024, doi: 10.1109/ACCESS.2024.3455410.

[6] S. G. E. Gökstorp, S. Katsikeas, and P. Johnson, "Machine learning for cybersecurity: A comprehensive literature review," *ACM Comput. Surv.*, vol. 58, no. 9, 2026.

[7] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in *Proc. IEEE Symp. Secur. Privacy (S&P)*, 2010, pp. 305–316.

[8] A. Al Siam *et al.*, "Securing the unseen: A comprehensive exploration review of AI-powered models for zero-day attack detection," *Expert Syst.*, vol. 43, no. 3, 2026.

[9] B. Kitchenham, "Procedures for performing systematic reviews," Keele Univ., Keele, U.K., Tech. Rep. TR/SE-0401, 2004.

[10] W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult, "Toward open set recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1757–1772, 2013.

[11] A. Bendale and T. E. Boult, "Towards open set deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 1563–1572.

[12] D. Arp *et al.*, "Dos and don'ts of machine learning in computer security," in *Proc. USENIX Secur. Symp.*, 2022, pp. 3971–3988.

[13] J. R. Quinlan, "Induction of decision trees," *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, 1986.

[14] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[15] Z. Dai, L. Y. Por, Y.-L. Chen, J. Yang, C. S. Ku, R. Alizadehsani, and P. Pławiak, "An intrusion detection model to detect zero-day attacks in unseen data using machine learning," *PLOS ONE*, vol. 19, Art. no. e0308469, 2024, doi: 10.1371/journal.pone.0308469.

[16] A. Touré, Y. Imine, A. Semnont, T. Delot, and A. Gallais, "A framework for detecting zero-day exploits in network flows," *Comput. Netw.*, vol. 248, Art. no. 110476, 2024, doi: 10.1016/j.comnet.2024.110476.

[17] Y. Ban *et al.*, "APSO-CNN-SE: An adaptive convolutional neural network approach for IoT intrusion detection," *Comput., Mater. Continua*, vol. 81, no. 1, 2024.

[18] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, 2008, pp. 413–422.

[19] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Comput.*, vol. 13, no. 7, pp. 1443–1471, 2001.

[20] M. Roopak, S. Parkinson, G. Y. Tian, Y. Ran, S. Khan, and B. Chandrasekaran, "An unsupervised approach for the detection of zero-day distributed denial of service attacks in Internet of Things networks," *IET Netw.*, vol. 13, no. 5–6, pp. 513–527, 2024, doi: 10.1049/ntw2.12134.

[21] M. A. Uddin, S. Aryal, M. R. Bouadjenek, M. Al-Hawawreh, and M. A. Talukder, "usfAD based effective unknown attack detection focused IDS framework," *Sci. Rep.*, vol. 14, Art. no. 29103, 2024, doi: 10.1038/s41598-024-80021-0.

[22] S. S. Tripathy, B. Behera, D. S. Kollie Jr., R. C. Barik, R. Priyadarshi, and R. Ranjan, "Enhancing anomaly-based zero-day attack detection framework using CNN-driven feature extraction and OC-SVM," *Sci. Rep.*, vol. 16, Art. no. 27164, 2026, doi: 10.1038/s41598-026-55170-z.

[23] M. Li, L. Luo, K. Xiao, G. Wang, and Y. Wang, "Adaptive semi-supervised algorithm for intrusion detection and unknown attack identification," *Appl. Sci.*, vol. 15, no. 4, Art. no. 1709, 2025, doi: 10.3390/app15041709.

[24] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: An ensemble of autoencoders for online network intrusion detection," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, 2018.

[25] H. Hindy, R. Atkinson, C. Tachtatzis, J.-N. Colin, E. Bayne, and X. Bellekens, "Utilising deep learning techniques for effective zero-day attack detection," *Electronics*, vol. 9, no. 10, Art. no. 1684, 2020, doi: 10.3390/electronics9101684.

[26] X. Yang, Z. Zhao, L. Wang *et al.*, "A variational autoencoder attention fusion model for zero-day exploit behavior detection VAE-AttNet," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 38, Art. no. 852, 2026, doi: 10.1007/s44443-026-01245-w.

[27] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2014.

[28] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 5998–6008.

[29] A. S. Almuflih *et al.*, "Securing IoT devices with zero day intrusion detection system using binary snake optimization and attention based bidirectional gated recurrent classifier," *Sci. Rep.*, vol. 14, Art. no. 29238, 2024, doi: 10.1038/s41598-024-80255-y.

[30] D. Krishnan, S. Singh, and V. Sugumaran, "Explainable AI for zero-day attack detection in IoT networks using attention fusion model," *Discover Internet Things*, vol. 5, Art. no. 83, 2025, doi: 10.1007/s43926-025-00184-8.

[31] J. Fang and C. Xie, "Unknown intrusion traffic detection method based on unsupervised learning and open-set recognition," *Sci. Rep.*, vol. 15, Art. no. 17001, 2025, doi: 10.1038/s41598-025-01084-1.

[32] A. Alsaleh, "A novel intrusion detection model of unknown attacks using convolutional neural networks," *Comput. Syst. Sci. Eng.*, vol. 48, no. 2, pp. 431–449, 2024, doi: 10.32604/csse.2023.043107.

[33] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, 2002.

[34] F. Alshammari and A. Alsaleh, "Smart intrusion detection model to identify unknown attacks for improved road safety and management," *Sci. Rep.*, vol. 15, Art. no. 18584, 2025, doi: 10.1038/s41598-025-03604-5.

[35] J. Zhang, Q. Wang, Z. Cai, K. Wang, and L. Zhu, "Unknown attack detection method based on deep metric learning," *Concurrency Comput., Pract. Exper.*, vol. 37, Art. no. e70193, 2025, doi: 10.1002/cpe.70193.

[36] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 1126–1135.

[37] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 4077–4087.

[38] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. ICML Deep Learn. Workshop*, 2015.

[39] Y. Yan, Y. Yang, F. Shen, M. Gao, and Y. Gu, "Meta learning-based few-shot intrusion detection for 5G-enabled industrial internet," *Complex Intell. Syst.*, 2024, doi: 10.1007/s40747-024-01388-1.

[40] N. Ijaz, S. Ullah Jan, and I. Koo, "Evaluating meta-learning strategies for zero-day intrusion detection under data scarcity," *Sci. Rep.*, 2026, doi: 10.1038/s41598-026-50116-x.

[41] J. F. Cevallos M. *et al.*, "NERO: NEural algorithmic reasoning for zeRO-day attack detection in the IoT: A hybrid approach," *Comput. Secur.*, vol. 142, Art. no. 103898, 2024, doi: 10.1016/j.cose.2024.103898.

[42] K. Alam, M. F. Monir, M. J. Hossain, M. S. Uddin, and M. T. Habib, "Adaptive defense: Zero-day attack detection in NIDS with deep reinforcement learning," *IEEE Access*, vol. 13, pp. 116345–116361, 2025, doi: 10.1109/ACCESS.2025.3585445.

[43] M. M. Hossain, S. D. Turja, S. Tasnim, M. F.-U.-A. Juboraj, and M. I. Hossain, "A cross-dataset based zero-day intrusion detection system by integrating Siamese network and reinforcement learning," *ICT Express*, vol. 12, no. 3, pp. 752–757, 2026, doi: 10.1016/j.icte.2026.05.001.

[44] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl. (CISDA)*, 2009, pp. 1–6.

[45] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems," in *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, 2015, pp. 1–6.

[46] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. Int. Conf. Inf. Syst. Secur. Privacy (ICISSP)*, 2018, pp. 108–116.

[47] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy," in *Proc. Int. Carnahan Conf. Secur. Technol. (ICCST)*, 2019, pp. 1–8.

[48] C. Kolias, G. Kambourakis, A. Stavrou, and S. Gritzalis, "Intrusion detection in 802.11 networks: Empirical evaluation of threats and a public dataset," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 184–208, 2016.

[49] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," *Sustain. Cities Soc.*, vol. 72, Art. no. 102994, 2021.

[50] E. C. P. Neto *et al.*, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," *Sensors*, vol. 23, no. 13, Art. no. 5941, 2023.

[51] G. Engelen, V. Rimmer, and W. Joosen, "Troubleshooting an intrusion detection dataset: The CICIDS2017 case study," in *Proc. IEEE Secur. Privacy Workshops (SPW)*, 2021, pp. 7–12, doi: 10.1109/SPW53761.2021.00009.

[52] M. Sarhan, S. Layeghy, and M. Portmann, "Towards a standard feature set for network intrusion detection system datasets," *Mobile Netw. Appl.*, vol. 27, no. 1, pp. 357–370, 2022.

[53] T. Ohtani, R. Yamamoto, and S. Ohzahata, "IDAC: Federated learning-based intrusion detection using autonomously extracted anomalies in IoT," *Sensors*, vol. 24, no. 10, Art. no. 3218, 2024, doi: 10.3390/s24103218.

[54] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Int. Conf. Artif. Intell. Statist. (AISTATS)*, 2017, pp. 1273–1282.

[55] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 4765–4774.

[56] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Comput. Surv.*, vol. 46, no. 4, Art. no. 44, 2014.

[57] B. E. Strom *et al.*, "MITRE ATT&CK: Design and philosophy," MITRE Corp., McLean, VA, USA, Tech. Rep., 2018.

[58] National Institute of Standards and Technology, "Artificial Intelligence Risk Management Framework (AI RMF 1.0)," NIST AI 100-1, Gaithersburg, MD, USA, Jan. 2023.

[59] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[60] A. Vassilev, A. Oprea, A. Fordyce, H. Anderson, X. Davies, and M. Hamin, "Adversarial machine learning: A taxonomy and terminology of attacks and mitigations," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, NIST AI 100-2e2025, Mar. 2025, doi: 10.6028/NIST.AI.100-2e2025.