Athens Journal of Sciences 2026, 13: 1-33
https://doi.org/10.30958/ajs.X-Y-Z

End-to-End Evaluation of AI, Agentic AI, and Autonomous Cybersecurity Systems

*By Paolina Centonze**

This paper evaluates artificial intelligence (AI), Agentic AI, and autonomous AI as complete cybersecurity workflows. The paper makes three major contributions: (1) a hierarchical role-and-assurance taxonomy and cross-family synthesis that maps statistical analytics, deep learning, graph neural networks, large language models, and Agentic AI to distinct cybersecurity lifecycle roles; (2) an end-to-end, complete-workflow evaluation framework that integrates precision, recall, F1, area under the receiver operating characteristic curve (AUROC), latency, cost, drift, robustness, unsafe-action rates, escalation, rollback, and auditability; and (3) an empirical synthesis that identifies where modern AI delivers measurable gains over earlier analytics and defines the human-governed safeguards required for bounded, reversible autonomy. Accuracy alone can conceal delay, cost, tool misuse, policy violations, and irreversible actions. Evidence shows F1 = 97.36%, AUROC = 99.59%, and 0.0476 ms latency; drift adaptation increased F1 from 0.60 to 0.86; policy-bounded remediation reduced violations from 12.18% to 5.82%; and layered defenses left 8.4% residual indirect prompt-injection vulnerability. Modern AI performs best on heterogeneous, relational, and multi-step tasks; autonomy should expand only when incremental workflow benefit is measured and failures remain detectable, containable, attributable, and reversible.

Keywords: *cybersecurity, machine learning, graph neural networks, large language models, agentic AI*

Introduction

Cybersecurity has become one of the most critical challenges facing modern organizations due to the rapid growth of cloud computing, Internet of Things devices, artificial intelligence, and interconnected digital infrastructures. As cyber threats continue to increase in scale and sophistication, traditional security approaches are no longer sufficient to analyze the massive volumes of security data generated daily (Denning 1987, Lunt 1993).

Over the past four decades, cybersecurity analytics have evolved from Statistical Analytics and Machine Learning to more advanced approaches such as Deep Learning, Large Language Models (LLMs), Graph Neural Networks (GNNs), and Agentic AI (Cortes and Vepnik 1995, Breiman 2001, Cover and Hart 1967, LeCun et al. 2015, Goodfellow et al. 2016). Each generation has introduced new capabilities for threat detection, pattern recognition, contextual reasoning, relationship modeling, and automated decision-making.

*Associate Professor, Iona University, USA.

LLMs have enhanced cybersecurity through natural language understanding and threat intelligence analysis (Vaswani et al. 2017, Ferrag et al. 2025, Zhang et al. 2025), while GNNs have enabled relationship-aware detection of complex attacks by modeling interactions among users, devices, applications, and threat actors (Kipf and Welling 2017, Hamilton et al. 2017, Alshehri et al. 2025). More recently, Agentic AI has introduced autonomous reasoning, planning, memory, and tool utilization capabilities that support advanced threat hunting, incident response, and Security Operations Center (SOC) automation (Lazer et al. 2026, Kim et al. 2026, Forough et al. 2026).

This paper presents a comparative analysis of these major data science methodologies, examining their strengths, limitations, explainability, scalability, and autonomy. The study also explores emerging research directions, including GNN–LLM fusion, secure Agentic AI, autonomous SOCs, and self-healing cyber defense systems that may shape the next generation of intelligent cybersecurity solutions (Bai et al. 2023, Touvron et al. 2023, Pan et al. 2023, Wei et al. 2026, Lazer et al. 2026, Kim et al. 2026, Forough et al. 2026).

The paper makes three major contributions: (1) a hierarchical role-and-assurance taxonomy and cross-family synthesis that maps statistical analytics, deep learning, graph neural networks, large language models, and Agentic AI to distinct cybersecurity lifecycle roles; (2) an end-to-end, complete-workflow evaluation framework that integrates precision, recall, F1, area under the receiver operating characteristic curve (AUROC), latency, cost, drift, robustness, unsafe-action rates, escalation, rollback, and auditability; and (3) an empirical synthesis that identifies where modern AI delivers measurable gains over earlier analytics and defines the human-governed safeguards required for bounded, reversible autonomy. The empirical synthesis covers detection, cyberattack prediction, penetration testing, incident response, and autonomous remediation. This analysis is important because prior model-centered comparisons can report high accuracy while overlooking failed investigations, unsafe tool calls, policy violations, residual prompt-injection risk, and irrecoverable actions. The resulting findings support a hybrid architecture in which autonomy is justified by incremental end-to-end benefit and bounded by human-governed controls.

Methodology

Review Design and Scope

This study uses a structured narrative review rather than a meta-analysis. Three research questions guide the synthesis: the first research question asks under which cybersecurity task conditions do AI-based approaches outperform statistical analytics and conventional machine learning; the second asks which architectural mechanisms account for the observed performance differences; and the third asks what evidence and controls are required before Agentic AI or autonomous AI can execute consequential response actions? The unit of analysis is the functional role performed in a cybersecurity workflow rather than the model label in isolation. The scope covers foundational statistical inference, conventional machine learning, deep

representation learning, graph representation learning, foundation-model reasoning, agentic orchestration, and bounded autonomous execution. Literature published from 1987 through July 2026 was considered, with priority given to peer-reviewed primary studies, machine-verifiable benchmarks, and technically substantive surveys. Preprints were retained only for rapidly developing agentic topics where peer-reviewed evidence remains limited.

Search Strategy and Eligibility

Searches were conducted across IEEE Xplore, the ACM Digital Library, Scopus, Web of Science, and arXiv. Search strings combined cybersecurity terms (cybersecurity, intrusion detection, malware, threat intelligence, attack path, lateral movement, security operations center) with method terms (statistical anomaly detection, machine learning, deep learning, graph neural network, large language model, and agentic AI). Backward and forward citation tracing supplemented database searches. Included sources described a cybersecurity application, architecture, evaluation, or material limitation and provided enough technical detail to support comparison. Duplicates, purely promotional material, studies outside cybersecurity, and sources lacking identifiable methods or evidence were excluded. Because the purpose is conceptual synthesis, the review does not claim systematic-review completeness and does not pool heterogeneous results.

Comparative Dimensions and Evidence Rules

The dimensions were selected to reflect both technical evaluation and deployment decisions: operational role; data and labeling requirements; detection rate, false-positive rate, precision, recall, F1 score, and area under the receiver operating characteristic curve when reported; inference latency; training and memory cost; scalability; robustness to concept drift and adversarial manipulation; explainability; human-intervention requirements; autonomy; and response risk. No weights were assigned and no overall score was calculated. Reported metrics are interpreted only within their original datasets and tasks because benchmark composition, class imbalance, preprocessing, attack coverage, and validation procedures differ. Qualitative labels summarize recurring evidence; they are not substitutes for head-to-head testing.

Hierarchical Taxonomy and Role-and-Assurance Framework

The conceptual model separates four nested layers. Statistical and conventional machine-learning methods provide foundational inference; deep learning supplies representation learning; graph neural networks are graph-structured deep-learning architectures; large language models are foundation models that may serve as reasoning and interface components; and agentic AI is an orchestration layer that can combine all of the preceding methods with memory, policies, and tools. These layers overlap by design. The framework then maps each layer to cybersecurity lifecycle roles and requires evidence and controls proportional to the consequence

of its output. A detector may be evaluated primarily on false positives and recall, whereas an autonomous containment action also requires authorization boundaries, reversibility, auditability, and fail-safe behavior.

Table 1. *Hierarchical Role-and-Assurance Framework for Cybersecurity Analytics*

Layer	Included approaches	Primary lifecycle roles	Evidence and assurance emphasis
Foundational inference	Statistical analytics; conventional machine learning	Telemetry baselining, anomaly detection, classification, prioritization	Recall, false-positive rate, calibration, drift, interpretable decision boundaries
Representation learning	Deep learning; GNNs as graph-structured deep models	Feature learning, malware/traffic classification, entity correlation, attack-path analysis	Task-specific validation, robustness, latency, compute cost, graph/data quality, explainability
Foundation-model reasoning	LLMs, often combined with retrieval or graph context	Threat-intelligence synthesis, investigation support, explanation, reporting, decision support	Factual grounding, prompt-injection resistance, privacy, provenance, human verification
Autonomous orchestration	Agentic AI combining models, memory, policies, and tools	Multi-step investigation, response planning, bounded containment, remediation coordination	Least privilege, authorization gates, sandboxing, reversibility, audit logs, deterministic controls, accountability

Source: Developed by the author as a conceptual synthesis of the reviewed literature (Denning 1987, Lunt 1993, Cortes and Vapnik 1995, Breiman 2001, Cover and Hart 1967, LeCun et al. 2015, Goodfellow et al. 2016, Alazab et al. 2011, Tavallaee et al. 2009, Sharafaldin et al. 2018, Xu et al. 2025, Kipf and Welling 2017, Hamilton et al. 2017, Alshehri et al. 2025, Vaswani et al. 2017, Ferrag et al. 2025, Zhang et al. 2025, Bai et al. 2023, Touvron et al. 2023, Pan et al. 2023, Wei et al. 2026, Lazer et al. 2026, Kim et al. 2026, Forough et al. 2026).

End-to-End Evaluation Framework

Evaluation Unit and Workflow Boundary

The evaluation unit is the complete operational workflow: telemetry intake, detection, enrichment, entity and event correlation, investigation, decision support, authorization, action, verification, rollback or recovery, and audit. A model-level F1 score cannot establish workflow effectiveness if alerts are not investigated, recommended actions are unsafe, or remediation cannot be verified. Each experiment should therefore specify the initial evidence, tools and permissions, human intervention points, termination condition, and machine-verifiable success criteria. End-to-end outcomes should include case-completion rate, mean time to detect, mean time to investigate, mean time to respond, evidence completeness, remediation success, rollback success, and the fraction of cases escalated to humans.

Detection and Ranking Metrics

Precision measures the proportion of positive predictions that are correct; recall measures the proportion of relevant attacks detected; and F1 provides their harmonic mean. AUROC evaluates ranking across decision thresholds, but precision-recall curves and per-class results remain essential under severe class imbalance. Studies should report confusion matrices, macro and weighted averages, confidence intervals, dataset prevalence, and threshold-selection procedures. For multi-stage workflows, these measures should be calculated not only for the detector but also for evidence retrieval, incident classification, root-cause identification, and remediation verification.

Latency, Cost, Drift, and Robustness

Latency should be reported at the detector, model-call, tool-call, authorization, action, and end-to-end case levels using median, p95, and p99 values. Cost should include tokens, model and tool charges, processor and accelerator time, peak memory, storage, network calls, and analyst minutes per completed case. Drift evaluation should use chronological splits and report the change in precision, recall, F1, and AUROC over time, drift-detection delay, recovery time, and labeling budget. Robustness testing should include unseen attacks, corrupted or missing telemetry, adversarial examples, poisoned retrieval, compromised tools, prompt injection, privilege-boundary tests, and failure of dependent services.

Unsafe-Action Rate and Human-Governed Autonomy

For Agentic AI, an unsafe action is any proposed or executed operation that violates policy, exceeds authorization, causes avoidable disruption, exposes protected data, destroys evidence, cannot be reversed as required, or proceeds without a mandated approval. Unsafe-action rate should be reported separately for proposals and executions, together with severity, blast radius, detection time, blocked-action rate, escalation rate, rollback success, and residual harm. This measure is more informative than task success alone: an agent that resolves incidents quickly but violates change-management rules is not operationally superior. Human oversight is part of the evaluated architecture, not an external exception.

Table 2. *Reported Detection, Efficiency, Drift, and Robustness Metrics*

Study / workflow	Precision / recall	F1 / AUROC	Latency / cost	Drift / robustness finding
Hybrid federated intrusion detection system (IDS) (Baidar et al. 2025)	Reported as high and balanced; exact values not stated in the abstract	F1 97.36%; AUROC 99.59% on UNSW-NB15 binary	0.0476 ms/sample on test hardware	Drift tolerance discussed; no longitudinal degradation rate reported
NetGuard generative active adaptation (Gupta et al. 2025)	Per-class detection evaluated; aggregate P/R not reported in abstract	Overall F1 0.60 to 0.86; rare-attack F1: 0.001 to 0.30, 0.04 to 0.50, 0.00 to 0.71	Labeling effort reduced; monetary/compute cost not reported in abstract	End-to-end Internet service provider and simulated evaluation explicitly targets drift and imbalance
GeoDyG-LLM graph-text prediction (Wei et al. 2026)	Not reported in available summary	F1 0.888	Not reported	Evidence specific to inter-state cyberattack data; cross-domain robustness not established
Cybersecurity AI Benchmark agent evaluation (Sanz-Gómez et al. 2025)	Not applicable to all tasks	Task success: ~70% knowledge, 20-40% multi-step attack/defense, 22% robotic targets	Not reported	Model-scaffold pairing produced up to 2.6x performance variation

Source: Reported results from the cited studies (Wei et al. 2026, Sanz-Gómez et al. 2025, Gupta et al. 2025, Baidar et al. 2025). Not reported indicates that a value was unavailable in the cited study summary and was not inferred.

Complete-Workflow and Safety Findings

The workflow evidence in Table 3 demonstrates why detection quality and autonomous safety must be reported together. PatchWeaver evaluates remediation under explicit change-management constraints and reports policy violations, misses, dwell time, admission latency, escalation, and resources rather than a model score alone (Li and Cao 2026). LogInject evaluates the complete path from adversarial log ingestion to harmful model output and then measures residual risk after layered defenses (Karanjai et al. 2026). The retrieval-barrier study accounts for attack construction cost, retrieval, and exfiltration in a multi-agent workflow (Chang et al. 2026). These designs more closely represent operational risk than isolated question-answering benchmarks.

Table 3. *Complete-Workflow Performance and Unsafe-Action Evidence*

Workflow	End-to-end performance	Latency / cost	Unsafe-action or robustness result	Governance implication
PatchWeaver autonomous Kubernetes remediation (Li and Cao 2026)	0.33% miss rate; 88.7% immediate blocks; 11.0% escalations	Mean dwell 0.84 s vs. 96.4 s toolchain; p99 admission 31 ms; <1 processor core and <1 gigabyte memory	Step violations 4.73% vs. 9.95%; episode violations 5.82% vs. 12.18%; 79.1% fewer step violations than LLM agent; p99 episode 16.8% vs. 51.3%	Typed policies, simulation, replanning, and escalation bound autonomy
LogInject LLM security-log analysis (Karanjai et al. 2026)	12,847 logs; 2,569 adversarial samples; four attack objectives	Not reported	Up to 88.2% and 83.4% average attack success; layered defense reduced attacks 90.4%, leaving 8.4% residual vulnerability	Input filtering alone is insufficient; combine filtering, prompt hardening, output validation, and human oversight
Retrieval-aware indirect prompt injection (Chang et al. 2026)	Near-100% malicious-fragment retrieval across 11 benchmarks and 8 embedding models; >80% Secure Shell key exfiltration in a multi-agent workflow	Attack construction as low as $0.21 per target query	High end-to-end compromise despite natural user query; evaluated defenses did not prevent malicious retrieval	Separate untrusted retrieval from secrets and tools; require provenance and release controls
SecRespond incident response (Wang et al. 2026)	23 models on 10 compromised cloud ranges; no model completed all detection and remediation requirements on any range	Not reported	Agents found alert-exposed problems but missed silent intrusions and complete verified remediation	Human investigation and approval remain necessary for high-impact response

Source: Reported results from the cited studies (Li and Cao 2026, Karanjai et al. 2026, Chang et al. 2026, Wang et al. 2026). Rates use each study's definitions and should not be pooled across workflows.

Agentic Security Architecture and Research Priorities

Indirect prompt-injection defense requires treating retrieved text, logs, emails, tickets, and tool output as untrusted data rather than instructions. Promising controls include content provenance, instruction-data separation, retrieval isolation, causal or typed information flow, tool-call authorization, output validation, and adversarial workflow testing. Trustworthy memory requires authenticated writes, provenance,

retention limits, conflict detection, poisoning scans, scoped retrieval, and the ability to inspect and delete state. Trustworthy tools require identity binding, version and integrity verification, explicit schemas, allow-listed arguments, sandboxing, timeout, and rate limits, and independent validation of consequential results.

Least-privilege execution should issue short-lived, task-scoped credentials and deny agents ambient authority. Irreversible actions should be replaced with staged, reversible operations wherever possible: quarantine before deletion, canary rollout before fleet deployment, snapshot before modification, and automatic rollback when verification fails. Multi-agent systems additionally require bounded delegation, authenticated inter-agent messages, shared-state integrity, budget and depth limits, loop detection, disagreement handling, and circuit breakers that prevent one agent's error from cascading through the team.

Auditability requires immutable records of prompts, retrieved evidence, memory reads and writes, model and policy versions, tool arguments, approvals, actions, observations, and rollback outcomes. Accountability requires a named human or organizational owner for the policy, deployment, authorization boundary, and incident review. Research should compare fully autonomous, human-on-the-loop, and human-in-the-loop configurations under the same workload. Autonomy should expand only when the incremental benefit over a constrained baseline is statistically and operationally significant, unsafe-action severity remains within a declared tolerance, and failures are detectable, containable, attributable, and reversible.

Threats to Validity

The cited studies use different datasets, tasks, baselines, model versions, and success criteria. Precision, recall, F1, AUROC, task success, miss rate, policy violation, and attack success are not interchangeable and are not pooled here. Several 2026 agentic studies are recent or prepublication results and require independent replication. Hosted-model behavior and cost can change without notice. Benchmark environments may omit enterprise noise, legacy dependencies, human coordination, and adaptive adversaries. The framework therefore treats the reported values as task-specific evidence and emphasizes transparent omissions, chronological validation, workflow-level measurement, and prospective testing in the target environment.

Statistical Analytics and Machine Learning Approaches

Introduction

Cybersecurity now requires more than fixed thresholds and isolated classifiers. Cloud-scale telemetry, encrypted and polymorphic malware, rapidly changing attack paths, unstructured threat intelligence, and machine-speed adversaries create volumes and relationships that previous data science pipelines cannot fully represent. Artificial intelligence addresses this gap by learning features from raw data, reasoning across text and graphs, retaining investigative context, and coordinating tools across the detection-response lifecycle (LeCun et al. 2015, Goodfellow et al. 2016, Alazab et al. 2011, Tavallaee et al. 2009, Sharafaldin et al. 2018, Xu et al. 2025, Kipf and Welling

2017, Hamilton et al. 2017, Alshehri et al. 2025, Vaswani et al. 2017, Ferrag et al. 2025, Zhang et al. 2025, Bai et al. 2023, Touvron et al. 2023, Pan et al. 2023, Wei et al. 2026, Lazer et al. 2026, Kim et al. 2026, Forough et al. 2026).

The central transition is from analytics that answer a bounded question to AI systems that assemble evidence and pursue an operational goal. Statistical analytics can identify deviation from a baseline, and conventional machine learning can classify a prepared feature vector. Deep learning can discover representations directly from traffic, binaries, and logs; graph neural networks can model relationships and attack paths; large language models can interpret reports, code, and analyst questions; Agentic AI can plan, invoke tools, evaluate intermediate results, and revise its approach. Autonomous AI extends the loop into bounded containment, remediation, and recovery. This functional expansion explains why modern AI can outperform previous approaches on complex, heterogeneous, and multi-step tasks.

Statistical Analytics represents the earliest generation of cybersecurity intelligence systems. Denning's Intrusion Detection Model introduced the concept of detecting suspicious behavior by comparing current activities against expected statistical profiles (Denning 1987). Early systems monitored login activity, network traffic, file access, and authentication failures to identify anomalies that might indicate malicious behavior. These approaches offered simplicity, low computational overhead, and high interpretability. However, they often suffered from high false-positive rates, limited adaptability, and difficulty detecting sophisticated attacks, making them less effective against modern cyber threats (Lunt 1993).

Transition from Statistical Analytics to Machine Learning

As attack techniques became more sophisticated, cybersecurity researchers recognized that manually defined statistical rules could not adequately capture the complex patterns of these attacks. Machine Learning emerged as a solution by enabling systems to learn patterns directly from data rather than relying exclusively on predefined rules. Machine Learning algorithms can identify nonlinear relationships among variables, improve detection accuracy, and adapt to changing threat environments. This transition marked a significant evolution in cybersecurity analytics, laying the foundation for modern intelligent defense systems (Cortes and Vapnik 1995).

Support Vector Machines (SVM)

Support Vector Machines (SVMs), introduced by Cortes and Vapnik, are widely used in cybersecurity for intrusion detection, malware classification, phishing detection, and user behavior analytics (Cortes and Vapnik 1995). SVMs classify data by identifying an optimal decision boundary between classes, such as benign and malicious activities. Their strong performance in high-dimensional datasets makes them particularly effective for cybersecurity applications. However, SVMs can be computationally expensive, require careful parameter tuning, and may struggle to scale in large and complex cyber environments.

Random Forests

Random Forests, introduced by Breiman, are ensemble learning methods that combine multiple decision trees to improve classification performance and reduce overfitting (Breiman 2001). Each decision tree independently evaluates cybersecurity features, and the final classification is determined through majority voting among all trees. Random Forests have become particularly popular in cybersecurity due to their ability to handle large numbers of features and their relatively strong interpretability. Random Forests are commonly used for intrusion detection, malware classification, threat intelligence analysis, behavioral analytics, and risk assessment. The advantages of Random Forests are high classification accuracy, robustness against overfitting, the ability to handle heterogeneous data, and feature importance analysis. Because Random Forests provide insight into which features contribute most significantly to classification decisions, they offer greater explainability than many Deep Learning approaches. The limitations of Random Forests include large memory requirements, reduced performance on extremely large datasets, and limited ability to model sequential behaviors. Despite these limitations, Random Forests remain among the most widely deployed Machine Learning techniques in operational cybersecurity environments.

K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN), proposed by Cover and Hart, is one of the simplest yet most effective Machine Learning algorithms (Cover and Hart 1967). KNN classifies new observations based on the labels of their nearest neighboring examples within a feature space. The underlying assumption is that similar observations tend to belong to the same class. In cybersecurity, KNN can identify malicious activities by comparing current behavior with previously observed attack patterns. Some of the Cybersecurity Applications of KNN are anomaly detection, intrusion detection, malware classification, and user behavior analytics. Advantages of the KNN simple implementation are no explicit training phase, effective for small and medium-sized datasets, intuitive decision-making process. Limitations of KNN are computationally expensive during prediction, sensitive to noisy data, and performance degradation in large-scale environments. Because KNN requires comparisons with all stored observations, scalability becomes a major challenge in modern enterprise cybersecurity systems.

Comparative Analysis of Traditional Machine Learning Approaches

Table 4 summarizes the characteristics of Statistical Analytics, SVM, Random Forests, and KNN.

Table 4. *Comparison of Statistical Analytics and Traditional Machine Learning Methods*

Method	Strengths	Limitations	Typical Cybersecurity Applications
Statistical Analytics	Simple, interpretable, efficient	High false positives	IDS, anomaly detection
SVM	High accuracy, strong generalization	Scalability challenges	Malware detection, IDS
Random Forests	Robust, explainable, accurate	Memory requirements	Threat classification
KNN	Simple, intuitive	Slow prediction, scalability issues	Behavioral analytics

Source: Developed by the author based on the literature review (Cover and Hart 1967).

Scalability Challenges

One of the most significant limitations of traditional Machine Learning approaches involves scalability. Modern organizations generate millions of network flows per day, billions of log entries, and thousands of daily security alerts. Algorithms such as KNN and SVM often struggle to process these volumes efficiently. Random Forests offer improved scalability but still face limitations when confronted with continuously evolving cyber environments. These challenges motivated the development of Deep Learning approaches capable of automatically learning complex feature representations and handling significantly larger datasets.

Deep Learning for Cybersecurity

Introduction

The increasing sophistication of cyber threats and the exponential growth of cybersecurity data have exposed limitations in traditional Machine Learning approaches. While algorithms such as Support Vector Machines (SVMs), random forests, and K-Nearest Neighbors (KNNs) significantly improved threat detection capabilities, they often required extensive feature engineering and struggled to capture complex nonlinear relationships within large-scale cybersecurity datasets (LeCun et al. 2015, Goodfellow et al. 2016). Deep Learning emerged as a transformative approach capable of automatically learning hierarchical feature representations directly from raw data. Advances in computational power, graphics processing units, and large-scale datasets enabled deep neural networks to achieve remarkable success across multiple cybersecurity applications, including intrusion detection, malware classification, phishing detection, anomaly detection, and behavioral analytics (Xu et al. 2025). Unlike traditional Machine Learning methods that rely heavily on manually crafted features, Deep Learning architectures automatically identify meaningful patterns from large datasets. This capability has made Deep Learning one of the most widely adopted technologies in modern cybersecurity research and operational security systems.

Evolution of Deep Learning in Cybersecurity

The resurgence of Neural Networks around 2012 marked the beginning of widespread Deep Learning adoption across numerous domains. Cybersecurity researchers quickly recognized the potential of Deep Learning for detecting sophisticated attacks hidden within massive volumes of network traffic and system logs. The evolution of Deep Learning in cybersecurity can be divided into three phases:

Phase 1: Basic Neural Network Adoption (2012–2015). Early applications focused on malware classification, spam detection, Intrusion detection, and behavioral analytics.

Phase 2: Specialized Deep Learning Architectures (2015–2020). Researchers began adopting Convolutional Neural Networks (CNNs), Long Short-Term Memory Networks (LSTMs), Autoencoders, and Recurrent Neural Networks (RNNs). These architectures significantly improved detection performance across multiple cybersecurity tasks.

Phase 3: Deep Learning-Based Intrusion Detection Systems (2020–Present). Recent research has focused on developing Deep Learning-Based Intrusion Detection Systems (deep-learning-based intrusion detection systems) capable of analyzing large-scale network traffic in real time while detecting increasingly sophisticated cyber threats (Xu et al. 2025).

Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are among the most successful Deep Learning architectures for cybersecurity applications. Originally developed for image recognition, CNNs have proven highly effective for analyzing structured cybersecurity data. CNNs utilize convolutional layers to automatically extract features from input data. This capability eliminates the need for extensive manual feature engineering and allows the model to learn relevant characteristics directly from network traffic, malware binaries, and system logs. Cybersecurity Applications of CNNs have been applied to malware detection, network intrusion detection, phishing detection, network traffic classification, and botnet detection. For example, malware binaries can be transformed into grayscale images and analyzed using CNN architectures. Research has demonstrated that CNNs can successfully identify malware families and variants by recognizing structural patterns within binary code. The advantages of CNNs are automatic feature extraction, high classification accuracy, effective pattern recognition, and reduced dependence on manual feature engineering. The limitations of CNNs include the need for large training datasets, High computational cost, Limited temporal analysis capabilities, and explainability challenges. Despite these limitations, CNNs remain among the most widely used Deep Learning architectures in cybersecurity.

Long Short-Term Memory Networks (LSTMs)

Many cybersecurity events occur in sequences over time. Traditional neural networks often struggle to capture long-term dependencies within sequential data. Long Short-Term Memory (LSTM) networks address this limitation through

specialized memory mechanisms. LSTMs are particularly effective for analyzing network traffic flows, authentication sequences, user behavior patterns, attack progression events, and security logs. Because cyberattacks often unfold across multiple stages, LSTMs are well-suited for detecting complex attack chains. Cybersecurity applications of LSTMs are commonly used for intrusion detection, Advanced Persistent Threat (APT) detection, insider threat detection, user behavior analytics, and network anomaly detection. For example, an attacker may initially compromise a workstation, move laterally through the network, escalate privileges, and eventually access sensitive resources. LSTMs can model these temporal dependencies and identify suspicious attack sequences.

Table 5. *Long Short-Term Memory Networks (LSTMs): Advantages and Limitations in Cybersecurity*

Aspect	Summary
Advantages of LSTMs	Temporal pattern recognition; Sequential attack analysis; Long-term dependency modeling; Improved behavioral analytics
Limitations of LSTMs	Computational complexity; Long training times; Difficulty explaining predictions; Dependence on large datasets

Source: Developed by the author based on the literature review.

Autoencoders for Anomaly Detection

Autoencoders represent another important class of Deep Learning models used extensively in cybersecurity. Unlike CNNs and LSTMs, autoencoders are typically trained using unsupervised learning techniques. An Autoencoder consists of two primary components encoder and the decoder. The encoder compresses input data into a lower-dimensional representation, while the decoder reconstructs the original input. When trained on normal behavior, autoencoders learn efficient representations of legitimate activities. Abnormal or malicious activities typically produce higher reconstruction errors, enabling anomaly detection. Cybersecurity applications of autoencoders are widely used for zero-day attack detection, network anomaly detection, insider threat detection, fraud detection, and behavioral analytics (Alazab et al. 2011). Because autoencoders do not require labeled attack data, they are particularly useful for identifying previously unseen threats.

Table 6. *Autoencoders: Advantages and Limitations in Cybersecurity*

Aspect	Summary
Advantages of Autoencoders	Unsupervised learning; Effective anomaly detection; Reduced dependence on labeled data Zero-day threat identification
Limitations of Autoencoders	False positives; Reconstruction sensitivity; Explainability challenges; Model tuning complexity

Source: Developed by the author based on the literature review.

Deep Learning-Based Intrusion Detection Systems (DL-IDS)

Deep Learning-Based Intrusion Detection Systems (deep-learning-based intrusion detection systems) represent one of the most active areas of cybersecurity research. These systems integrate Deep Learning architectures with traditional IDS frameworks to improve detection performance. Recent surveys have demonstrated that deep-learning-based intrusion detection systems frequently outperform traditional Machine Learning approaches in terms of detection accuracy, precision, recall, and false-positive reduction. Xu et al. (2025) conducted a comprehensive survey of Deep Learning-Based Intrusion Detection Systems and reported substantial performance improvements across multiple benchmark datasets (Xu et al. 2025). These architectures commonly combine CNNs, LSTMs, autoencoders, and Hybrid Deep Learning models to improve overall detection effectiveness. The key benefits of DL-IDS are automatic feature extraction, improved scalability, enhanced detection accuracy, and better detection of complex attacks. Some of the challenges are computational requirements, explainability limitations, data quality dependence, and model drift.

Table 7. *Deep Learning: Strengths and Limitations in Cybersecurity*

Aspect	Summary
Strengths of Deep Learning	Automatic feature extraction; High detection accuracy; Complex pattern recognition; Scalability for large datasets; Improved threat detection performance
Limitations of Deep Learning	Large training data requirements; Computational expense; Explainability challenges; Model drift; Difficulty interpreting decisions

Source: Developed by the author based on the literature review.

Despite these challenges, Deep Learning remains one of the most effective technologies for cybersecurity analytics.

Summary

Deep Learning transformed cybersecurity by enabling automatic feature extraction, high detection accuracy, and scalable analysis of large security datasets. Architectures such as CNNs, LSTMs, and Autoencoders have significantly improved malware detection, intrusion detection, anomaly detection, and behavioral analytics. Deep Learning-Based Intrusion Detection Systems (deep-learning-based intrusion detection systems) consistently outperform traditional Machine Learning methods on benchmark datasets such as NSL-KDD and CICIDS2017 (Tavallaee et al. 2009, Sharafaldin et al. 2018, Xu et al. 2025). Despite challenges related to explainability, computational cost, and large data requirements, Deep Learning remains a core technology in modern cyber defense. Its limitations in understanding unstructured text motivated the emergence of Large Language Models (LLMs) for cybersecurity reasoning and threat intelligence analysis.

Large Language Models and Cybersecurity (LLMs)

Introduction to Large Language Models

Large Language Models (LLMs) are a decisive advance over prior cybersecurity analytics because they operate directly on the unstructured information that dominates investigations: advisories, tickets, code, commands, malware reports, policies, emails, and threat-intelligence narratives (Vaswani et al. 2017, Ferrag et al. 2025, Zhang et al. 2025). Conventional models normally require these materials to be converted into task-specific features and labels. LLMs instead provide a reusable semantic layer for extraction, correlation, explanation, question answering, and report generation. When grounded with retrieval and connected to verified tools, they reduce analyst switching costs and make accumulated security knowledge available inside the workflow rather than in a separate search process.

Evolution of LLMs in Cybersecurity

Natural Language Processing has long been used in cybersecurity for tasks such as spam filtering, phishing detection, and document classification. The introduction of Transformer architectures revolutionized language processing (Vaswani et al. 2017), leading to foundation models such as GPT-4, Claude, Gemini, and Llama (Touvron et al. 2023, Anil et al. 2023, OpenAI 2023). In addition, the Transformer architectures in 2017 significantly improved contextual understanding and led to the development of Large Language Models (LLMs) capable of advanced reasoning and language generation (Vaswani et al. 2017). The evolution of LLMs in cybersecurity can be categorized into three major phases. Between 2018 and 2021, LLM applications focused primarily on information retrieval and classification tasks such as security document classification, threat report categorization, vulnerability tagging, and knowledge management. From 2022 to 2024, capabilities expanded to generative security assistance, including threat intelligence summarization, vulnerability explanation, malware description generation, and security report creation. Since 2024, LLMs have evolved into cybersecurity reasoning systems integrated with autonomous agents, enabling applications such as SOC copilots, threat hunting assistants, vulnerability assessment agents, and autonomous incident analysis. This progression reflects the transition from simple text-processing tools to intelligent decision-support systems capable of assisting cybersecurity operations and investigations.

Several Large Language Models have emerged as leading platforms for cybersecurity applications. Models such as GPT-4, Claude, Gemini, and Llama have significantly expanded the capabilities of cybersecurity analytics and decision support systems (Brown et al. 2020, Open AI 2023, Lewis et al. 2020). GPT models are widely used for threat intelligence summarization, malware analysis, vulnerability assessment, incident response assistance, and security report generation due to their strong reasoning capabilities (Ferrag et al. 2025). Claude excels at processing lengthy documents and maintaining contextual consistency, making it valuable for threat intelligence analysis, compliance reviews, and incident investigations. Gemini emphasizes multimodal reasoning and enterprise integration, supporting threat

intelligence correlation, vulnerability analysis, and security operations workflows. Llama provides organizations with an open-source alternative that enables private deployment of cybersecurity-focused LLMs, making it particularly attractive for environments with strict privacy, compliance, and data sovereignty requirements (Touvron et al. 2023).

Cybersecurity Applications of LLMs

Large Language Models (LLMs) have significantly expanded cybersecurity capabilities across multiple domains. They support threat intelligence analysis by summarizing reports, extracting indicators of compromise, correlating information, and mapping adversary activities to frameworks such as MITRE ATT&CK (Ferrag et al. 2025, Zhang et al. 2025). LLMs also enhance Security Operations Centers (SOCs) through alert explanation, incident prioritization, and automated report generation. In vulnerability management and malware analysis, they assist by explaining vulnerabilities, assessing risks, interpreting malicious code behavior, and generating analyst-ready summaries. Additionally, organizations increasingly leverage LLMs to develop personalized security awareness training, phishing simulations, and interactive cybersecurity education programs, improving both operational efficiency and workforce preparedness.

Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation (RAG) is a significant advancement that enhances Large Language Models by combining their reasoning capabilities with external knowledge sources (Ferrag et al. 2025). RAG also improves factual accuracy and reduces hallucinations by combining language models with external knowledge retrieval systems (Significant Gravitas 2023). Unlike traditional LLMs that rely solely on training data, RAG enables access to real-time threat intelligence, organizational knowledge bases, and up-to-date cybersecurity information. This approach improves factual accuracy, reduces hallucinations, enhances explainability, and provides more reliable recommendations. As a result, many modern cybersecurity copilots and AI-assisted security platforms use RAG architectures to support threat analysis and decision-making.

Strengths of LLMs for Cybersecurity

Large Language Models offer several advantages for cybersecurity, including natural language understanding, contextual reasoning, knowledge integration, and enhanced human-AI interaction. They can process threat reports, security advisories, vulnerability disclosures, incident reports, and research papers while combining information from multiple sources such as threat intelligence feeds, vulnerability databases, and historical incidents. These capabilities enable more effective cybersecurity analysis, improve analyst productivity, and reduce workload through automated summarization, documentation, and reporting, allowing security professionals to focus on higher-level decision-making tasks.

Limitations of LLMs

Despite their significant advantages, Large Language Models face several important limitations. Hallucinations can generate plausible but incorrect cybersecurity information, potentially introducing errors into threat intelligence workflows (Ferrag et al. 2025). LLMs are also vulnerable to prompt injection attacks, where malicious instructions manipulate model behavior. Because they rely on historical training data, LLMs may provide outdated or inaccurate cybersecurity knowledge. Additionally, processing sensitive security information through external LLM services can create privacy, confidentiality, and compliance concerns. Finally, the quality and reliability of LLM outputs depend heavily on prompt design, making effective prompt engineering essential for accurate cybersecurity analysis.

Comparison with Previous Data Science Approaches

Table 8. *Comparison of LLMs with Previous Methodologies*

Feature	Statistical Analytics	Machine Learning	Deep Learning	LLMs
Natural Language Understanding	No	No	Limited	Excellent
Threat Intelligence Analysis	Limited	Limited	Moderate	Excellent
Explainability	High	Moderate	Low	Moderate
Human Interaction	Limited	Limited	Limited	Excellent
Contextual Reasoning	Low	Moderate	Moderate	High
SOC Assistance	Limited	Limited	Moderate	Excellent

Source: Developed by the author based on the literature review.

Table 8 illustrates that LLMs introduce capabilities largely absent from previous generations of cybersecurity analytics.

Future Research Directions

Several promising research directions are emerging for Large Language Models in cybersecurity. Domain-specific cybersecurity LLMs may outperform general-purpose models by leveraging specialized training data (Ferrag et al. 2025). Improving explainability and transparency remains a critical challenge for broader adoption. Research on GNN–LLM fusion seeks to combine graph-based reasoning with natural language understanding to enhance threat intelligence correlation, attack-path analysis, adversary tracking, and vulnerability prioritization (Wei et al. 2026). Additionally, LLMs are increasingly being integrated into Agentic AI systems as reasoning engines that support autonomous investigation, planning, and response (Lazer et al. 2026, Kim et al. 2026). These advances may ultimately enable Autonomous Security Operations Centers capable of coordinating multiple AI agents and cybersecurity tools with minimal human intervention.

Summary

Large Language Models have significantly transformed cybersecurity by enabling advanced language understanding, contextual reasoning, and automated knowledge extraction. They are widely used for threat intelligence analysis, malware investigation, vulnerability management, and Security Operations Center (SOC) assistance, improving both analyst productivity and decision-making (Ferrag et al. 2025, Zhang et al. 2025). Although challenges such as hallucinations, prompt injection attacks, privacy concerns, and knowledge reliability remain, advances in Retrieval-Augmented Generation (RAG), cybersecurity-specific models, and Agentic AI integration continue to enhance their effectiveness. As cybersecurity increasingly depends on unstructured information, LLMs are expected to become core components of next-generation cyber defense platforms and autonomous security operations.

Graph Neural Networks for Cybersecurity (GNNs)

Introduction

As cyber environments become increasingly interconnected, traditional cybersecurity analytics often struggle to capture the complex relationships among users, devices, applications, vulnerabilities, and threat actors. Graph Neural Networks (GNNs) address this challenge by learning directly from graph structures that represent network topologies, communication flows, attack paths, and threat intelligence relationships (Kipf and Welling 2017, Hamilton et al. 2017, Alshehri et al. 2025). Unlike traditional Machine Learning, Deep Learning, and Large Language Models, GNNs explicitly model interactions among connected entities, making them highly effective for Advanced Persistent Threat (APT) detection, lateral movement analysis, attack-path modeling, and threat intelligence correlation. As cyberattacks increasingly involve sequences of interconnected activities, GNNs are emerging as a critical technology for next-generation cybersecurity systems (Alshehri et al. 2025, Pan et al. 2023).

Graph Representation Learning

Graph Representation Learning forms the foundation of Graph Neural Networks by transforming graph entities into low-dimensional vector representations that preserve structural and relational information. In cybersecurity, graphs typically consist of nodes representing users, devices, servers, applications, domains, Internet Protocol addresses, malware families, and threat actors, while edges represent relationships such as network communications, authentication events, file transfers, and threat intelligence associations. This approach enables cybersecurity systems to capture both individual entity characteristics and the relationships among connected entities, providing valuable context for threat detection and analysis.

Graph Convolutional Networks (GCNs)

Graph Convolutional Networks (GCNs), introduced by Kipf and Welling (2017), are among the most influential GNN architectures. Unlike traditional neural networks that analyze independent observations, GCNs aggregate information from neighboring nodes through iterative message passing, allowing each node to learn increasingly rich contextual representations. In cybersecurity, GCNs have been successfully applied to intrusion detection, malware detection, insider threat analysis, network traffic classification, and Advanced Persistent Threat (APT) detection. Research shows that GCNs can uncover hidden attack relationships often missed by traditional Machine Learning and Deep Learning approaches (Alshehri et al. 2025).

GraphSAGE

Although GCNs achieve strong performance, they face scalability challenges when applied to large enterprise networks. GraphSAGE, introduced by Hamilton et al. (2017), addresses this limitation by sampling neighborhoods rather than processing entire graphs. This approach improves scalability, supports inductive learning, handles previously unseen nodes, and reduces computational requirements. In cybersecurity, GraphSAGE has been applied to dynamic threat detection, large-scale enterprise monitoring, threat intelligence correlation, and user behavior analytics, making it well-suited for continuously evolving cyber environments.

Advanced Persistent Threat (APT) Detection

Advanced Persistent Threats are among the most sophisticated cyberattacks, often involving multiple stages including initial compromise, privilege escalation, lateral movement, persistence, and data exfiltration. Traditional detection systems frequently analyze these stages independently, making it difficult to identify the broader attack campaign. GNNs address this challenge by modeling attack activities as interconnected graphs, allowing analysts to examine entire attack paths rather than isolated events. Research indicates that GNNs significantly improve APT detection accuracy by capturing relationships among attack stages and identifying patterns that would otherwise remain hidden (Alshehri et al. 2025).

Lateral Movement Detection

Lateral movement is a critical stage of many cyberattacks in which adversaries move across systems after gaining initial access. Examples include credential theft, remote desktop protocol abuse, Pass-the-Hash attacks, and internal reconnaissance. Because these activities involve relationships among users, devices, and authentication paths, graph-based analysis is particularly effective. GNNs can model privilege escalation chains, user-to-device relationships, and network communications, enabling earlier detection of suspicious movement patterns before attackers reach critical assets.

Threat Intelligence Correlation

Threat intelligence data is often collected from multiple sources, including security vendors, government agencies, open-source intelligence feeds, and internal security teams. Correlating information across these sources is challenging but essential for effective cyber defense. GNNs support threat intelligence fusion by representing entities such as threat actors, malware families, domains, Internet Protocol addresses, vulnerabilities, and exploits as interconnected graph structures. Through graph analysis, GNNs can uncover hidden relationships among threat indicators, improve attribution efforts, and enhance situational awareness and threat prediction capabilities (Alshehri et al. 2025).

Dynamic Graph Challenges

Despite their advantages, GNNs face challenges related to the dynamic nature of cybersecurity environments. Networks continuously evolve as new devices are added, users change roles, services are deployed, and threat actors modify tactics. Static graph models can quickly become outdated under these conditions. Dynamic Graph Neural Networks seek to address this issue by continuously updating graph representations over time; however, maintaining accurate and efficient representations of rapidly changing cyber environments remains an active area of research.

Scalability Limitations

Scalability remains one of the primary obstacles to widespread GNN adoption in cybersecurity. Large organizations may contain millions of devices, billions of connections, and continuously evolving authentication and communication graphs. As the graph size increases, challenges related to memory consumption, training complexity, graph sampling overhead, and real-time processing become more significant. Although architectures such as GraphSAGE improve scalability, researchers continue exploring distributed GNN architectures, graph compression techniques, incremental learning methods, and real-time graph analytics to support enterprise-scale cybersecurity deployments.

Strengths and Limitations of GNNs

Graph Neural Networks offer several important advantages, including relationship-aware threat detection, attack-path modeling, improved APT detection, enhanced lateral movement analysis, threat intelligence fusion, and contextual reasoning. These capabilities enable cybersecurity systems to model complex interactions among entities more effectively than traditional ML, DL, and LLM approaches. However, GNNs also face challenges related to scalability, dynamic graph complexity, computational requirements, explainability, and deployment complexity, all of which remain active research areas.

Summary

Graph Neural Networks represent a major advancement in cybersecurity analytics by enabling AI systems to learn directly from graph structures and exploit relationships among cybersecurity entities. Through graph representation learning, GCNs, and GraphSAGE architectures, GNNs have demonstrated strong performance in APT detection, lateral movement analysis, insider threat detection, and threat intelligence correlation. Although challenges related to scalability, dynamic graph management, and explainability remain, GNNs are increasingly viewed as foundational technologies for next-generation cyber defense systems. Furthermore, emerging research suggests that integrating GNNs with Large Language Models may further enhance cyber threat prediction, reasoning, and autonomous security operations.

Agentic AI for Cybersecurity

Introduction

Agentic AI is the most important architectural shift examined in this review because it converts AI from a passive predictor into an operational participant. Earlier models emit a score, class, embedding, or answer and then stop. An agent persists across steps: it interprets a goal, forms and revises a plan, calls security tools, evaluates evidence, records state, and escalates or acts within policy (Lazer et al. 2026, Kim et al. 2026, Yao et al. 2023, Deng et al. 2024). In cybersecurity, this enables continuous threat hunting, automated evidence collection, adaptive vulnerability analysis, investigation across multiple systems, and coordinated response. Autonomous AI extends the same loop into execution and outcome verification, creating the possibility of machine-speed defense when permissions, safeguards, and rollback are engineered correctly.

What Is Agentic AI?

Agentic AI refers to Artificial Intelligence systems that can autonomously pursue goals through iterative cycles of perception, reasoning, planning, action, and learning (Yao et al. 2023, Lazer et al. 2026). Unlike traditional AI systems that generate a single response, Agentic AI continuously evaluates its environment and adapts its behavior. In cybersecurity, these agents can monitor networks, investigate suspicious activities, correlate threat intelligence, and recommend mitigation actions with minimal human supervision.

Core Components of Agentic AI

Agentic AI is built upon four key capabilities: reasoning, planning, memory, and tool utilization. Reasoning enables agents to analyze problems and make informed decisions, while planning transforms objectives into actionable workflows. Memory allows agents to retain historical knowledge, previous investigations, and learned procedures. Modern Agentic AI architectures often leverage Large Language Models

as reasoning engines while incorporating planning, memory, and tool usage capabilities (OWASP Foundation 2025, Wang et al. 2024). Tool utilization extends agent capabilities by enabling interaction with security information and event management platforms, vulnerability scanners, threat intelligence feeds, malware sandboxes, and other cybersecurity technologies.

Agentic AI Architectures

Several architectures support Agentic AI systems. The ReAct (Reasoning + Acting) framework combines reasoning and action in continuous cycles that allow agents to adapt to evolving situations (Lewis et al. 2020). Multi-agent architectures further enhance capabilities by enabling specialized agents, such as threat hunting, malware analysis, vulnerability assessment, and incident response agents, to collaborate and achieve broader cybersecurity objectives (Sanz-Gómez et al. 2025).

Agentic AI Applications in Cybersecurity

Agentic AI is transforming cybersecurity through autonomous threat hunting, SOC automation, malware analysis, and vulnerability management. These systems can analyze logs, detect anomalies, correlate indicators of compromise, assess vulnerabilities, and generate investigation reports. Within Security Operations Centers, Agentic AI supports alert triage, incident prioritization, response recommendations, and automated investigations, significantly improving operational efficiency and reducing analyst workload (Kim et al. 2026).

Security Vulnerabilities of Agentic AI

Agentic AI introduces system-level risks that exceed ordinary model error. Direct or indirect prompt injection can manipulate an agent through user input, retrieved threat intelligence, tickets, emails, or web content. Hallucinated conclusions may become unsafe tool calls; compromised tools can falsify observations; persistent memory can be poisoned; excessive permissions can enable privilege escalation; and one erroneous agent can propagate failures through a multi-agent workflow (Lazet et al. 2026, Forough et al. 2026, OWASP Foundation 2025, Wang et al. 2024). Non-deterministic behavior complicates reproducibility, incident reconstruction, and responsibility assignment. Accordingly, production designs require least-privilege credentials, authenticated tools and data provenance, isolated execution, allow-listed actions, policy checks, rate and scope limits, immutable audit trails, rollback mechanisms, continuous monitoring, and mandatory human approval for irreversible or high-impact actions.

When Agents Handle Secrets

A growing concern involves agents accessing sensitive information such as credentials, application programming interface keys, authentication tokens, and confidential documents. Research by Forough et al. (2026) highlight the risks associated with autonomous systems handling privileged information and emphasizes the need for

secure memory architectures, confidential computing environments, and strong access-control mechanisms to protect sensitive assets (Forough et al. 2026).

Agentic AI Security Operations Centers

One of the most promising applications of Agentic AI is the development of Autonomous Security Operations Centers. These systems can automate alert triage, threat intelligence enrichment, malware analysis, incident investigation, report generation, and response orchestration. Benefits include faster investigations, improved consistency, enhanced scalability, and reduced analyst fatigue, representing a major step toward autonomous cyber defense.

Future Research Directions

Future research is expected to focus on explainable Agentic AI, secure autonomous agents, and the integration of Large Language Models and Graph Neural Networks (Wei et al. 2026, Lazet et al. 2026, Wang et al. 2024). Combining graph reasoning, natural language understanding, external tools, and persistent memory may create more capable cyber defense platforms. Researchers are also exploring self-healing cyber systems capable of detecting attacks, diagnosing problems, applying mitigations, and restoring services with minimal human intervention.

Strengths and Limitations

Agentic AI offers broad workflow coverage through multi-step reasoning, memory, tool integration, and coordination. That breadth should not be confused with superior detection accuracy, reliability, or deployment readiness. Agents generally depend on underlying statistical or learned detectors and may add latency, cost, attack surface, and decision variance. Their appropriate use is therefore risk-tiered: low-impact enrichment and drafting may be automated; consequential containment or remediation should remain bounded by deterministic policy and human authorization until performance, security, and accountability are demonstrated in the target environment.

Summary

Agentic AI is best understood as an orchestration framework rather than a peer alternative to statistical models, deep networks, GNNs, or LLMs. It can connect these components across investigation and response, but present evidence does not justify calling it universally superior or fully deployment-ready. Its value depends on bounded authority, trustworthy inputs and tools, measurable workflow gains, and controls that keep failures reversible and attributable.

Comparative Data Science Analysis for Cybersecurity

Interpretation of the Comparison

The six labels describe different abstraction levels and operational functions. Deep learning is a subset of machine learning; GNNs are deep architectures specialized for relational data; many LLMs are deep foundation models; and an agent may invoke rules, classifiers, GNNs, LLMs, retrieval systems, and security tools. Consequently, a single low-to-very-high capability ladder is scientifically misleading. Table 9 compares the approaches by the work they perform, the evidence needed to evaluate that work, and the risks created when outputs are allowed to trigger actions.

Table 9. *Cybersecurity Analytics Role-and-Assurance Matrix*

Approach	Typical lifecycle role	Relevant indicators	Maturity / principal limitation	Human and safety boundary
Statistical analytics	Telemetry baselining; anomaly detection	False-positive rate, recall, latency, calibration	High maturity for bounded tasks; weak on complex or drifting behavior	Analyst validates alerts and tunes thresholds
Conventional machine learning	Classification; prioritization; prediction	Precision, recall, F1, AUROC, drift, inference cost	Mature for scoped tasks; depends on labels and features	Human owns deployment, drift review, and response
Deep learning	Representation learning; malware and traffic analysis	Task metrics, robustness, compute, data volume, latency	Operational but assurance varies; opaque and data intensive	Explainability and escalation controls required
GNNs	Entity correlation; lateral movement and attack-path analysis	Node/edge/subgraph metrics, recall, scalability, graph freshness	Emerging in many operational settings; graph construction is consequential	Analyst validates graph context and inferred relationships
LLMs	Threat-intelligence analysis; investigation and decision support	Factuality, citation accuracy, task success, latency, privacy, injection resistance	Rapidly developing; hallucination and prompt injection remain material	Grounding, provenance, privacy controls, and human verification
Agentic AI	Workflow orchestration; bounded investigation and response	End-to-end task success, time to detect/respond, unsafe-action rate, reversibility, audit coverage	Early-stage for consequential autonomy; broad attack surface and accountability gaps	Least privilege, policy gates, sandboxing, audit, rollback, human approval

Source: Developed by the author from the literature synthesis literature (Denning 1987, Lunt 1993, Cortes and Vapnik 1995, Breiman 2001, Cover and Hart 1967, LeCun et al. 2015, Goodfellow et al. 2016, Alazab et al. 2011, Tavallaee et al. 2009, Sharafaldin et al. 2018, Xu et al. 2025, Kipf and Welling 2017, Hamilton et al. 2017, Alshehri et al. 2025, Vaswani et al. 2017, Ferrag et al. 2025, Zhang et al. 2025, Bai et al. 2023, Touvron et al. 2023, Pan et al. 2023, Wei et al. 2026, Lazer et al. 2026, Kim et al. 2026, Forough et al. 2026, Lewis et al. 2020, Yao et al. 2023, Significant Gravitas 2023, OWASP Foundation 2025, Wang et al. 2024).

Operational Lifecycle Findings

The comparison shows complementarity rather than succession. Statistical and learned detectors operate closest to telemetry and classification. GNNs add relationship-aware correlation across entities and events. LLMs are strongest where unstructured text, explanation, and analyst interaction dominate. Agents coordinate multi-step work across these components, but they do not independently guarantee better detection. Containment, remediation, and recovery are qualitatively different from analysis because errors can disrupt services or destroy evidence; increasing action authority must therefore increase assurance and oversight.

Limits of Quantitative Comparison

Detection rate, precision, recall, F1, AUROC, latency, training cost, memory use, data requirements, robustness to concept drift, explainability, human intervention, and response risk are all relevant, but the reviewed studies do not report them under common conditions. Results based on NSL-KDD, CICIDS2017, proprietary enterprise logs, malware corpora, threat reports, or simulated agent tasks are not directly aggregable. Dataset age, class balance, leakage, preprocessing, attack taxonomy, threshold selection, and validation design can alter results materially. This review therefore reports the appropriate indicators and recurring tradeoffs without calculating a pooled effect or declaring a universal winner. A defensible selection requires head-to-head testing on the organization's data, workload, threat model, and risk tolerance.

Critical Synthesis

Modern AI is superior to earlier analytics when the task requires learned representations, heterogeneous evidence, natural-language context, relational reasoning, or multi-step tool use. The quantitative studies summarized above show large gains over direct-model baselines: +228.6% task completion for modular penetration testing, 83.3% versus 27.8% success in tool-driven binary analysis, and F1 = 0.888 for graph-text prediction (Wei et al. 2026, Deng et al. 2024, Liu et al. 2026). Previous methods remain preferable for stable, bounded decisions where low latency and transparent failure modes dominate. The strongest architecture is therefore AI-centered but hybrid: statistical controls and specialized classifiers supply dependable signals; deep, graph, and language models build context; and agents orchestrate investigation and carefully bounded action.

Future Research Directions

Introduction

The rapid evolution of Artificial Intelligence (AI), Machine Learning, Deep Learning, Graph Neural Networks (GNNs), Large Language Models (LLMs), and Agentic AI is fundamentally transforming cybersecurity. While current cybersecurity

systems have achieved significant advances in threat detection, vulnerability assessment, malware analysis, and incident response, future cyber defense environments will require higher levels of intelligence, adaptability, autonomy, explainability, and resilience. Cyber adversaries are increasingly employing AI-driven attack techniques, automated reconnaissance tools, advanced malware, and sophisticated social engineering campaigns. Consequently, future cybersecurity solutions must evolve beyond reactive defense mechanisms toward proactive and autonomous systems capable of predicting, detecting, responding to, and recovering from cyberattacks with minimal human intervention. Several emerging research directions have the potential to redefine cybersecurity over the next decade. These include GNN–LLM fusion architectures, Agentic AI Security Operations Centers (SOCs), autonomous malware analysis platforms, explainable cybersecurity AI, self-healing cyber infrastructures, confidential computing environments, and graph-enhanced multi-agent cyber defense systems.

GNN-LLM Fusion for Cybersecurity

Graph Neural Networks (GNNs) and Large Language Models (LLMs) offer complementary capabilities for cybersecurity. GNNs excel at modeling relationships among users, devices, applications, domains, and threat actors, while LLMs provide natural language understanding and contextual reasoning. Future GNN–LLM fusion architectures may improve cyberattack prediction, threat intelligence correlation, attack-path analysis, and vulnerability prioritization by combining graph reasoning with language-based explanations and decision support (Wei et al. 2026).

Agentic AI Security Operations Centers

Agentic AI has the potential to transform Security Operations Centers (SOCs) by automating alert triage, threat hunting, malware analysis, incident investigation, and response orchestration (Lazer et al. 2026, Kim et al. 2026). These autonomous systems may significantly reduce mean time to detect and mean time to respond while improving analyst productivity and operational scalability. Future research should focus on human-in-the-loop architectures that balance autonomy with oversight and accountability.

Autonomous Malware Analysis

Malware analysis remains a resource-intensive cybersecurity task. Agentic AI systems can automate malware execution, behavioral observation, indicator-of-compromise extraction, threat intelligence correlation, and report generation. Future platforms may integrate LLM reasoning and autonomous planning to create highly efficient malware investigation ecosystems capable of accelerating incident response and threat attribution.

Explainable AI for Cybersecurity

As AI systems gain greater autonomy, explainability becomes increasingly important. Security analysts must understand why alerts, classifications, and remediation actions are generated. Future Explainable AI (XAI) research will focus on interpretable Deep Learning models, explainable GNNs, transparent LLMs, and auditable Agentic AI workflows to improve trust and adoption of autonomous cybersecurity systems.

Self-Healing Cyber Systems

Self-healing cyber systems represent one of the most ambitious goals of cybersecurity research. These systems aim to automatically detect attacks, diagnose root causes, apply corrective actions, verify recovery, and restore normal operations without human intervention. By combining Deep Learning, GNNs, LLMs, and Agentic AI, future self-healing architectures may provide adaptive and resilient cyber defense capabilities.

Confidential Computing for AI-Driven Cybersecurity

As AI systems increasingly process sensitive security information, protecting data, memory, and communications becomes critical. Confidential Computing provides hardware-based protection for AI workloads, reducing risks associated with credentials, authentication tokens, security logs, and threat intelligence [24]. Future research will explore integrating confidential computing with Agentic AI, LLMs, and autonomous SOCs to improve trust, privacy, and security.

Graph-Enhanced Multi-Agent Cyber Defense

Future cybersecurity ecosystems are expected to employ multiple specialized AI agents collaborating across threat hunting, malware analysis, vulnerability management, incident response, and threat intelligence functions. By leveraging GNNs for shared situational awareness and coordinated decision-making, graph-enhanced multi-agent systems may improve threat correlation, attack-path analysis, scalability, and resilience against complex cyber threats.

Future Cybersecurity Ecosystem

The future of cybersecurity will likely involve the convergence of Statistical Analytics, Machine Learning, Deep Learning, GNNs, LLMs, and Agentic AI into unified defense architectures. These platforms will combine graph reasoning, natural language understanding, autonomous planning, explainable decision-making, confidential computing, and multi-agent collaboration to deliver increasingly intelligent and autonomous cyber defense capabilities.

Summary

The next generation of cybersecurity systems will be shaped by the integration of GNNs, LLMs, Agentic AI, Explainable AI, and Confidential Computing. Advances in GNN–LLM fusion, autonomous SOCs, self-healing cyber systems, and graph-enhanced multi-agent architectures promise to improve threat detection, investigation, response, and resilience. Although significant technical and governance challenges remain, these technologies collectively represent the future of intelligent and autonomous cyber defense.

Table 10. *Evolution of Data Science Methodologies and Future Directions in Cybersecurity*

Methodology	Approximate Time Period	Key Technologies	Key Contributions	Future Directions
Statistical Analytics	1980s–1990s	Statistical Models, Behavioral Profiling	First anomaly-based intrusion detection systems; established the foundation of cybersecurity analytics	Machine Learning-based detection
Machine Learning	1995–2015	Decision Trees, Random Forests, SVMs, Clustering	Automated classification and prediction; improved detection accuracy over statistical methods	Deep Learning and automated feature learning
Deep-learning intrusion detection	2010–Present	CNNs, RNNs, LSTMs, Autoencoders, Transformers	Automated feature learning, malware detection, anomaly detection, and zero-day attack identification	Explainable AI (XAI), GNNs, Agentic AI
Large Language Models (LLMs)	2022–Present	Transformer-Based Foundation Models	Threat intelligence analysis, SOC assistance, cyber reasoning, incident summarization, and security automation	RAG, GNN–LLM Fusion, Agentic AI
Graph Neural Networks (GNNs)	2018–Present	Graph Intelligence, Relational Learning, Graph Embeddings	Graph-based threat modeling, attack-path analysis, lateral movement detection, APT detection, and threat correlation	Explainable GNNs, Dynamic Graph Learning, LLM Integration
Graph-Text Fusion	2024–Present	GNN + LLM Fusion Architectures	Combines graph reasoning with language intelligence to improve cyber intelligence, threat prediction, and contextual understanding	Autonomous SOCs, Agentic AI, Multi-Agent Systems
Agentic AI	2025–Present	Autonomous AI Agents, Planning, Memory, Tool Use	Autonomous reasoning, decision making, workflow orchestration, and adaptive cyber defense	Multi-Agent Cyber Defense Systems, Autonomous SOCs

Methodology	Approximate Time Period	Key Technologies	Key Contributions	Future Directions
Agentic AI Security	2025–Present	Secure Agent Architectures, Agent Guardrails, Runtime Monitoring	Agent runtime security, supply-chain protection, adversarial resilience, and threat taxonomy development	Zero-Trust Agentic AI, Secure Runtime Architectures
Confidential Agentic AI	2025–Present	Confidential Computing, trusted execution environments, Secure Multi-Party Computing	Protection of secrets, credentials, memory, and agent communications through trusted execution environments	Trusted Multi-Agent Systems, Privacy-Preserving AI
Autonomous Cyber Defense Ecosystems	2027+ (Emerging Vision)	GNNs + LLMs + Agentic AI + Confidential Computing	Fully autonomous, explainable, machine-speed cyber defense ecosystems capable of continuous monitoring, reasoning, and response	Self-Healing Networks, Autonomous SOCs, Human–AI Collaborative Defense

Source: Developed by the author based on the literature synthesis.

Table 10 illustrates the evolution of cybersecurity analytics from Statistical Analytics to emerging Autonomous Cyber Defense Ecosystems. Each generation has increased the level of intelligence, contextual awareness, and automation in cyber defense. While Deep Learning, LLMs, and GNNs have significantly improved detection, reasoning, and threat analysis, recent advances in Graph-Text Fusion and Agentic AI are enabling more autonomous security operations. The convergence of these technologies is expected to drive the development of explainable, secure, and self-healing cyber defense systems.

Conclusion

Cybersecurity AI should be judged as an operational system rather than as an isolated model. Precision, recall, F1, and AUROC establish detection quality, but deployment decisions also require end-to-end latency, computational and monetary cost, resilience to drift, adversarial robustness, unsafe-action rates, escalation, rollback, recovery, and auditability.

The paper makes three major contributions: (1) a hierarchical role-and-assurance taxonomy and cross-family synthesis that maps statistical analytics, deep learning, graph neural networks, large language models, and Agentic AI to distinct cybersecurity lifecycle roles; (2) an end-to-end, complete-workflow evaluation framework that integrates precision, recall, F1, area under the receiver operating characteristic curve (AUROC), latency, cost, drift, robustness, unsafe-action rates, escalation, rollback, and auditability; and (3) an empirical synthesis that identifies where modern AI delivers measurable gains over earlier analytics and defines the human-governed safeguards required for bounded, reversible autonomy.

These contributions matter because isolated benchmarks can conceal workflow failure and operational harm. The findings show that modern AI provides its clearest

gains on heterogeneous, relational, unstructured, and multi-step tasks, while the agentic evidence also exposes prompt-injection compromise, incomplete remediation, and policy-violation risk. Consequently, the analysis supports hybrid and human-governed architectures: autonomy should expand only when its incremental benefit is measured against a constrained baseline and its failure modes are detectable, containable, attributable, and reversible.

The comparative analysis further indicates that no single methodology is sufficient to address all cybersecurity challenges. Instead, future cyber defense architectures will likely combine the strengths of multiple approaches, creating hybrid systems capable of leveraging statistical analysis, machine learning, graph intelligence, language understanding, and autonomous decision-making. This convergence represents a significant step toward intelligent and adaptive cybersecurity ecosystems.

Several promising research directions emerge from this study. These include GNN–LLM fusion for enhanced cyber situational awareness, Agentic AI Security Operations Centers (SOCs), autonomous malware analysis, Explainable AI (XAI), confidential computing for AI-driven security, graph-enhanced multi-agent defense systems, and self-healing cyber infrastructures capable of autonomously detecting, analyzing, mitigating, and recovering from cyberattacks. These research areas offer significant opportunities for interdisciplinary collaboration among cybersecurity, artificial intelligence, data science, and software engineering researchers.

Future work should establish shared, chronological, and adversarial workflow benchmarks that report precision, recall, F1, AUROC, p50/p95/p99 latency, cost per completed case, drift degradation and recovery, robustness under compromised inputs and tools, unsafe-action rates, escalation, and rollback success. Agentic research priorities include indirect prompt-injection defenses, trustworthy memory and tools, least-privilege execution, reversible actions, multi-agent failure containment, auditability, and clear accountability. The evidence supports hybrid and human-governed architectures: autonomy should expand only when its incremental benefit is measured against a constrained baseline and its failure modes are bounded.

References

Alazab M, Venkatraman S, Watters P, Alazab M (2011) Zero-Day Malware Detection Based on Supervised Learning Algorithms. *Journal of Universal Computer Science* 17(13): 1827–1840.

Alshehri SM, Sharaf SA, Molla RA (2025) Systematic Review of Graph Neural Network for Malicious Attack Detection. *Information* 16(6): 470.

Anil R, et al. (2023) *Gemini: A Family of Highly Capable Multimodal Models*. arXiv:2312.11805.

Bai Y, et al. (2023) *Constitutional AI: Harmlessness from AI Feedback*. arXiv:2212.08073.

Baidar R, Maric S, Abbas R (2025) *Hybrid Deep Learning-Federated Learning Powered Intrusion Detection System for IoT/5G Advanced Edge Computing Network*. arXiv:2509.15555.

Breiman L (2001) Random Forests. *Machine Learning* 45(1): 5–32.

Brown TB, et al. (2020) Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems (NeurIPS)* 33: 1877–1901.

Chang H, Bao E, Luo X, Yu T (2026) Overcoming the Retrieval Barrier: Indirect Prompt Injection in the Wild for LLM Systems. In *Proceedings of the 35th USENIX Security Symposium*, 2026.

Cortes C and Vapnik V (1995) Support-Vector Networks. *Machine Learning* 20(3): 273–297.

Cover TM, Hart PE (1967) Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory* 13(1): 21–27.

Deng G, Liu Y, Mayoral-Vilches V, Liu P, Li Y, Xu Y, et al. (2024) PentestGPT: Evaluating and Harnessing Large Language Models for Automated Penetration Testing. In *Proceedings of the 33rd USENIX Security Symposium*, 2024.

Denning DE (1987) An Intrusion-Detection Model. *IEEE Transactions on Software Engineering* 13(2): 222–232.

Ferrag MA, Tihanyi N, Debbah M, Lestable T (2025) Revolutionizing Cybersecurity with Large Language Models: A Comprehensive Survey. *IEEE Communications Surveys & Tutorials* 27(1): 1–39.

Forough J, Kogias M, Haddadi H (2026) *When Agents Handle Secrets: A Survey of Confidential Computing for Agentic AI*. arXiv:2605.03213.

Goodfellow I, Bengio Y, Courville A (2016) *Deep Learning*. Cambridge, MA, USA: MIT Press.

Gupta R, Liu S, Zhang R, Hu X, Wang X, Benkraouda H, et al. (2025) *Generative Active Adaptation for Drifting and Imbalanced Network Intrusion Detection*. arXiv:2503.03022.

Hamilton W, Ying Z, Leskovec J (2017) Inductive Representation Learning on Large Graphs. *Advances in Neural Information Processing Systems (NeurIPS)* 30.

Karanjai R, Lu Y, Madhavarao HH, Xu L, Shi W (2026) Context Contamination in LLM Analysis of Network Security Logs: Poison with Passive Prompt Injection and Mitigation Evaluation. In *Proceedings of the 35th USENIX Security Symposium*, 2026.

Kim J, Liu X, Wang Z, Qiu S, Li B, Guo W, et al. (2026) *The Attack and Defense Landscape of Agentic AI: A Comprehensive Survey*. arXiv:2603.11088.

Kipf TN, Welling M (2017) Semi-Supervised Classification with Graph Convolutional Networks. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2017.

Lazer SJ, Aryal K, Gupta M, Bertino E (2026) *A Survey of Agentic AI and Cybersecurity: Challenges, Opportunities and Use-case Prototypes*. arXiv:2601.05293.

LeCun Y, Bengio Y, Hinton G (2015) Deep Learning. *Nature* 521(7553): 436–444.

Lewis P, et al. (2020) Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Proceedings of NeurIPS*, 2020.

Li R, Cao S (2026) PatchWeaver: Risk-Bounded Autonomous Vulnerability Remediation Under Change-Management Policies. In *Proceedings of the 35th USENIX Security Symposium*, 2026.

Liu Y, Liu Z, Yu Z, Wang S, Jiang L, Nie S, et al. (2026) Towards Generality: Task-Adaptive Binary Analysis via Semantic Retrieval and Verifiable Reasoning. In *Proceedings of the 35th USENIX Security Symposium*, 2026.

Lunt TF (1993) A Survey of Intrusion Detection Techniques. *Computers & Security* 12(4): 405–418.

OpenAI (2023) *GPT-4 Technical Report*. arXiv:2303.08774.

OWASP Foundation (2025) *OWASP Top 10 for LLM Applications 2025*. OWASP Foundation.

Pan S, Cambria E, Wang C, Hooi M (2023) Graph Neural Networks in Artificial Intelligence: A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45(3): 289–305.

Sanz-Gómez M, Mayoral-Vilches V, Balassone F, Navarrete-Lozano LJ, Veas Chavez CRJ, del Mundo de Torres M (2025) *Cybersecurity AI Benchmark (CAIBench): A Meta-Benchmark for Evaluating Cybersecurity AI Agents*. arXiv:2510.24317.

Sharafaldin I, Lashkari AH, Ghorbani AA (2018) Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. In *Proceedings of 4th International Conference on Information Systems Security and Privacy (ICISSP)*, Funchal, Portugal, 2018, 108–116.

Significant Gravitas T (2023) *AutoGPT: An Autonomous GPT-4 Experiment*. GitHub Repository.

Tavallaee M, Bagheri E, Lu W, Ghorbani AA (2009) A Detailed Analysis of the KDD CUP 99 Data Set. In *Proceedings of IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA)*, Ottawa, Canada, 2009, 1–6.

Touvron H, et al. (2023) *LLaMA: Open and Efficient Foundation Language Models*. arXiv: 2302.13971.

Vaswani A, et al. (2017) Attention Is All You Need. *Advances in Neural Information Processing Systems (NeurIPS)* 30: 5998–6008.

Wang J, et al. (2024) A Survey of Large Language Model Based Multi-Agent Systems. arXiv:2402.01680.

Wang L, Chen B, Ding R, Xie P, Huang J, Liu Z, et al. (2026) *SecRespond: Benchmarking AI Agents for Real-World Post-Compromise Incident Response*. arXiv:2607.26791.

Wei H, Ferreira-Soriano M, Martin H, Stakhanova N, Burnap P (2026) Predicting Inter-State Cyberattacks with Graph-Text Fusion Using Graph Neural Networks and Large Language Models. *Engineering Applications of Artificial Intelligence* 165: 113465.

Xu Z, Wu Y, Wang S, Gao J, Qiu T, Wang Z, et al. (2025) *Deep Learning-Based Intrusion Detection Systems: A Survey*. arXiv:2504.07839.

Yao S, et al. (2023) *ReAct: Synergizing Reasoning and Acting in Language Models*. arXiv:2210.03629.

Zhang Y, Xu J, Li X, Zhao H (2025) Large Language Models in Cybersecurity: Applications, Challenges, and Future Directions. *ACM Computing Surveys* 58(4): 1–42.