DMPedia Lecture Notes in Multidisciplinary Research,
Volume 1, pp. 295-314, 2026

International Conference on Multidisciplinary Perspectives in Advanced Computing and Technology (IMPACT 2026)
G. B. Pant University of Agriculture and Technology, Uttarakhand, India. Jan. 10-11, 2026

Deep Learning Assisted Optimization Models for Reducing Network Application Vulnerabilities in an Organization

Kismat Chhillar[1], Saurabh Shrivastava[1] , Deepak Tomar[2] , Alok Verma[1]

[1]Dept of Mathematical Sciences and Computer Applications, Bundelkhand University, Jhansi, Uttar Pradesh, India

[2]Computer Center, Bundelkhand University, Jhansi, Uttar Pradesh, India

drkismatchhillar@gmail.com, dr.saurabh@bujhansi.ac.in, dr.deepak@bujhansi.ac.in, alokverma.bu@gmail.com

ABSTRACT

The growing interconnectedness and structural complexity of organizational networks have significantly amplified their exposure to application-level vulnerabilities. This intensifying threat landscape calls for the development of intelligent and proactive mitigation mechanisms. The present study introduces a deep learning assisted optimization framework designed to enhance the efficiency and adaptability of vulnerability reduction in networked applications. The framework integrates advanced neural architectures with multi-objective optimization strategies to achieve a systematic and data-driven mitigation process. Within the proposed framework, hybrid deep learning models autonomously extract latent vulnerability patterns from threat intelligence feeds and network log data. The optimization component dynamically prioritizes remediation actions based on the severity, exploitability, and potential impact of each vulnerability. The collaboration between predictive learning and optimization modules facilitates adaptive decision-making, minimizing downtime and improving the allocation of organizational resources during vulnerability management operations. Empirical evaluation using real-world datasets demonstrates significant improvements in detection precision, reduced false positive rates, and faster patch management cycles compared to traditional heuristic approaches. By combining deep learning inference with optimization analytics, this research contributes a scalable and intelligent solution that strengthens organizational resilience against evolving cyber threats, thereby advancing the pursuit of autonomous and adaptive vulnerability mitigation in enterprise environments.

Keywords: *Deep Learning, Optimization, Network Vulnerabilities, Cybersecurity, Threat Mitigation, Vulnerability Management, Neural Networks, Adaptive Security*

1. Introduction

1.1. Background on Increasing Vulnerability Trends in Enterprise Networks

The rapid digitalization of organizational infrastructures has substantially increased the complexity and exposure surface of enterprise networks, leading to a surge in security vulnerabilities that can be exploited by malicious actors. Modern enterprises rely extensively on interconnected applications, cloud-based services, and distributed computing, which collectively amplify systemic interdependencies and potential points of failure. As network applications evolve to deliver real-time services and scalability, they also inherit risks associated with software misconfigurations, outdated libraries, and inadequate security validation [1]. The continuous expansion of threat vectors, coupled with the sophistication of cyberattacks such as zero-day exploits, lateral movement intrusions, and AI-driven malware, poses a serious challenge to traditional defense mechanisms [2], [3]. Consequently, the

 ISBN: 978-81-993813-5-3

frequency and impact of security breaches have intensified, causing severe disruptions in business continuity, financial losses, and erosion of customer trust. This growing threat landscape underscores the urgent need for proactive, intelligent systems capable of autonomously detecting, predicting, and mitigating vulnerabilities before they are exploited [4], [5].

1.2. Limitations of Conventional Vulnerability Assessment and Patch Management

Traditional vulnerability assessment techniques, including signature-based scanning and rule-driven evaluation, struggle to keep pace with the dynamic and evolving nature of contemporary cyber threats [6]. These approaches primarily rely on predefined vulnerability databases and manual analysis, which often result in delayed identification of new exploits and misclassification of emerging risks. Moreover, conventional patch management frameworks frequently operate on static schedules and fail to account for contextual factors such as network topology, asset value, and operational dependencies. This leads to inefficient resource allocation and unnecessary system downtime during remediation cycles. The absence of automation and data-driven intelligence also limits the scalability and accuracy of these methods, leaving enterprises exposed to high-impact vulnerabilities that escape initial detection or prioritization [7]. Therefore, overcoming the rigidity and inefficiency of conventional approaches necessitates the integration of intelligent modelling techniques that can adapt continuously to threat dynamics and optimize mitigation strategies based on real-time organizational contexts.

1.3. Motivation for Integrating Deep Learning with Optimization Models

Deep learning has emerged as a transformative paradigm in cybersecurity due to its capacity to model complex, nonlinear patterns within high-dimensional data, making it ideal for detecting subtle vulnerabilities and behavioural anomalies in network traffic. By learning from vast and diverse datasets, deep neural networks can identify latent relationships among system attributes, vulnerability indicators, and threat behaviours that are often imperceptible to traditional analytical methods [8]. However, detection alone is insufficient; organizations must also dynamically prioritize and implement mitigation strategies in resource-constrained environments. This is where the integration of optimization models complements deep learning by guiding decision-making processes through quantitative evaluation of trade-offs among severity, cost, and urgency. The synergy between deep learning and optimization thus creates a unified framework that not only recognizes vulnerabilities but also determines the most effective remediation sequence, thereby improving both detection efficiency and operational resilience. Such integration represents a paradigm shift from reactive detection systems to intelligent, self-adaptive security management architectures.

1.4. Research Gaps Identified in Existing Literature

Although existing studies demonstrate the effectiveness of deep learning and optimization techniques individually within cybersecurity applications, there remains a clear gap in the development of unified frameworks that integrate both approaches for comprehensive vulnerability mitigation in organizational networks. Most current research has focused

primarily on intrusion detection, malware classification, and static vulnerability assessment, with limited exploration into dynamic vulnerability management and adaptive prioritization strategies. Consequently, the interplay between predictive detection and intelligent remediation remains underdeveloped in current literature. Moreover, few existing models adequately address the inherently multi-objective nature of vulnerability reduction, where competing factors such as patching latency, service continuity, and risk severity must be optimized simultaneously. Another notable limitation lies in the restricted generalizability of these models across heterogeneous network infrastructures and evolving vulnerability landscapes, often resulting in reduced applicability to real-world enterprise environments. These gaps underscore the need for an integrated deep learning–assisted optimization framework that delivers scalable, context-aware, and automated defense mechanisms to support adaptive vulnerability management in contemporary organizational networks.

1.5. Objectives and Contributions of This Study

The primary objective of this study is to develop and validate a deep learning–assisted optimization framework that enhances both the intelligence and operational efficiency of organizational vulnerability management. The research focuses on improving detection accuracy, reducing response and remediation times, and minimizing the recurrence of critical vulnerabilities through adaptive and data-driven modelling approaches. The proposed framework integrates hybrid deep learning architectures for advanced vulnerability detection with optimization algorithms that dynamically prioritize and schedule remediation tasks according to network conditions, resource constraints, and organizational goals. The major contributions of this work include the design of a context-aware architecture that unifies vulnerability assessment and optimization, the introduction of a multi-objective methodological model that jointly addresses risk severity, operational continuity, and patch timeliness, and the empirical validation of the framework's scalability and resilience within complex enterprise network environments. Collectively, these contributions advance the domain of AI-driven cybersecurity by bridging the existing divide between intelligent detection systems and adaptive remediation optimization, offering a comprehensive solution for proactive and autonomous vulnerability management.

1.6. Structure of the Remaining Paper

The remainder of this paper is structured as follows. Section 2 provides a comprehensive literature review highlighting advancements and limitations in deep learning applications and optimization models for network vulnerability mitigation. Section 3 presents the proposed methodology, describing the conceptual architecture, dataset preparation, model development process, and the mathematical formulation of the optimization component. Section 4 outlines the experimental setup, including system configuration, dataset characteristics, and implementation details. Section 5 discusses the results and analysis, evaluating the performance of the proposed framework against existing state-of-the-art methods. Thereafter, Section 6 concludes the paper by summarizing the key findings, emphasizing the contributions, and suggesting potential directions for future research in

advancing deep learning-assisted optimization for organizational cybersecurity management. Lastly, section 7 discusses about the future scope of the current research.

2. Literature Review

2.1. Overview of Traditional Vulnerability Scanning and Patch Prioritization

Traditional vulnerability scanning tools primarily rely on signature-based detection mechanisms and heuristic rules to identify known security flaws within organizational networks [9], [10]. These scanners perform regular assessments to detect exposed weaknesses by comparing system configurations and software versions against curated vulnerability databases such as CVE and NVD. Patch prioritization methods conventionally depend on static severity metrics, often using Common Vulnerability Scoring System (CVSS) scores to decide remediation urgency [11]. While these established approaches provide baseline defensive capabilities, they struggle with scalability and timeliness, especially in large-scale, dynamic enterprise environments. Limitations include high false-positive rates, inability to predict emerging vulnerabilities, and inadequate consideration of contextual factors such as asset criticality and network topology. These challenges underscore the need for more adaptive, data-driven techniques to accelerate and refine vulnerability recognition and patch deployment within complex organizational infrastructures.

2.2. Role of Machine Learning and AI in Security Vulnerability Management

Machine learning and artificial intelligence have demonstrated considerable promise in augmenting traditional vulnerability management processes by enabling automated detection, risk prediction, and intelligent prioritization [12]. Supervised and unsupervised learning models analyse historical vulnerability data, network traffic patterns, and system logs to uncover hidden security threats and anomalous behaviours that conventional tools may overlook [13], [14]. Techniques such as decision trees, support vector machines, and clustering algorithms have been employed to enhance early vulnerability recognition, reduce false alerts, and predict exploitability trends [15], [16]. Furthermore, AI-driven platforms facilitate predictive patch management by recommending optimal remediation schedules based on evolving threat intelligence and organizational risk exposure. However, many machine learning models face challenges in handling high-dimensional, imbalanced datasets characteristic of cybersecurity domains, and their interpretability remains limited, restricting operational adoption [17].

2.3. Review of Deep Learning Architectures Applied in Cybersecurity

Deep learning, a subset of machine learning, excels at modelling complex nonlinear relationships and hierarchical representations, making it particularly suitable for cybersecurity applications. Architectures such as Convolutional Neural Networks (CNNs) are effective in spatial pattern recognition, enabling detection of subtle structural vulnerabilities in code or network packets [18], [19]. Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks capture temporal dependencies in

sequential data, facilitating anomaly detection in time-series network logs and user behaviour analytics [20], [21]. Autoencoders, with their capacity for unsupervised feature learning, contribute to identifying unknown or zero-day vulnerabilities by learning compact representations of normal system states and signalling deviations as potential threats [22], [23]. Recent studies highlight the integration of hybrid models combining CNN and LSTM layers to achieve superior detection accuracies in dynamic network environments [24], [25]. Despite these advances, deep learning models often require extensive labelled datasets, are computationally intensive, and face difficulties in explainability, which pose challenges for real-time organizational deployment.

2.4. Optimization Techniques Used in Vulnerability Mitigation

Optimization methods have increasingly been applied in cybersecurity to address the multi-faceted challenge of vulnerability prioritization and remediation scheduling [26]. Genetic algorithms leverage evolutionary principles to explore large search spaces for optimal patch deployment sequences that balance risk reduction and operational costs [27]. Particle Swarm Optimization (PSO) algorithms simulate social behaviours to optimize defense resource allocations across network components [28], [29]. Reinforcement learning methods, particularly deep reinforcement learning, enable adaptive decision-making frameworks that learn optimal mitigation policies through interactions with dynamic threat environments and feedback on outcome effectiveness. Multi-objective optimization models consider conflicting goals such as minimizing vulnerability exposure, reducing patching downtime, and controlling remediation costs [30]. These approaches enhance organizational capacity to make data-driven, context-aware remediation choices that improve cybersecurity posture. However, many optimization models simplify real-world constraints or suffer from convergence issues, limiting their practical applicability in highly complex and volatile network settings.

2.5. Identified Limitations in Existing Models and Research Gaps

Despite notable progress, current literature reveals persistent limitations and gaps in vulnerability management frameworks that combine deep learning with optimization. Most existing approaches focus narrowly on either vulnerability detection or remediation but lack comprehensive integration of both under a unified adaptive framework. Current models often do not sufficiently handle real-time data streams or adapt to rapidly evolving vulnerability landscapes, resulting in delayed or suboptimal mitigation actions [31]. Issues related to model interpretability, trustworthiness, and robustness against adversarial attacks remain critical challenges limiting deployment in sensitive enterprise contexts. Additionally, scalability concerns and the need for privacy-preserving mechanisms in multi-tenant or distributed environments are under-addressed. There is a clear need for research that develops explainable, scalable, and context-aware deep learning-assisted optimization models capable of operating under realistic constraints and providing actionable threat mitigation strategies for organizational networks. Such approaches should include hybrid architectures, real-time adaptability, and rigorous validation in heterogeneous cyber environments.

3. Methodology

3.1. System Architecture

The proposed framework adopts a modular architecture that integrates deep learning–based vulnerability detection with an optimization engine for adaptive remediation. It is composed of four core components: data acquisition and pre-processing, feature extraction, a deep learning prediction model, and an optimization layer responsible for patch prioritization and scheduling. The data acquisition module gathers heterogeneous inputs from multiple sources, including network traffic logs, vulnerability databases, and system audit trails. The pre-processing module normalizes and transforms these raw data streams into structured feature representations suitable for model ingestion. The prediction module employs hybrid deep neural networks combining convolutional and recurrent layers to effectively capture both spatial and temporal patterns associated with vulnerabilities. The optimization component applies multi-objective algorithms to prioritize and schedule vulnerabilities by jointly minimizing exploit likelihood and operational disruption. The architecture incorporates real-time feedback mechanisms that continuously update and refine mitigation strategies based on changing network conditions and incoming threat intelligence. This adaptive design achieves a balance between predictive precision and operational efficiency, ensuring that vulnerability detection and remediation remain both timely and context-aware in complex enterprise environments.

3.2. Data Preparation

Data preparation constitutes a critical phase in the development of the proposed framework, involving the construction of a comprehensive dataset that captures network application states, historical vulnerabilities, and relevant contextual metadata. The dataset integrates information from organizational network logs, public vulnerability repositories such as the Common Vulnerabilities and Exposures (CVE) database, and real-time threat intelligence feeds to ensure wide and up-to-date coverage of potential attack vectors. The data cleaning process eliminates noise, redundancy, and inconsistencies, while feature engineering extracts meaningful attributes including protocol anomalies, patch histories, severity ratings, exploitability factors, and asset criticality scores. To mitigate the challenge of class imbalance between vulnerable and non-vulnerable instances, synthetic minority over-sampling techniques (SMOTE) are employed. The curated dataset is partitioned into training, validation, and testing subsets using stratified sampling to maintain representative class distribution. Feature normalization and dimensionality reduction methods, such as principal component analysis, are applied to enhance data quality and optimize model learning. These pre-processing steps contribute to faster convergence, improved prediction stability, and overall robustness of the deep learning models in vulnerability detection and prioritization.

3.3. Deep Learning Model

The core vulnerability detection mechanism employs a hybrid deep learning approach that integrates Convolutional Neural Networks (CNNs) for spatial pattern recognition with Long Short-Term Memory (LSTM) networks for modelling temporal correlations within

sequential network data. The CNN layers extract hierarchical representations from structured input vectors that encode network traffic behaviours and application code characteristics, enabling the identification of subtle anomaly signatures indicative of potential vulnerabilities. The LSTM layers capture temporal dependencies and patterns in the evolution of vulnerabilities, allowing the model to detect emerging threats and behavioural shifts that static classifiers often overlook. The architecture comprises multiple hidden layers augmented with dropout regularization and batch normalization to counter overfitting and enhance generalization. The model is trained through backpropagation using the Adam optimizer, with critical hyper parameters such as learning rate, batch size, and neuron configuration tuned according to validation performance metrics. The loss function integrates cross-entropy to ensure classification accuracy while incorporating custom penalty components that emphasize the correct identification of high-severity vulnerabilities. This integrative modelling framework significantly improves detection precision, minimizes false positive rates, and provides early warning capabilities essential for proactive vulnerability management in enterprise cybersecurity systems.

3.4. Optimization Phase

Following vulnerability identification, the optimization module governs the remediation process by formulating and solving a multi-objective optimization problem that balances risk reduction, patch deployment time, and operational continuity. The module incorporates a reinforcement learning mechanism that interacts with a simulated network environment to learn optimal policies for patch scheduling and execution. The state space represents key contextual factors such as vulnerability severity, exploit likelihood, asset criticality, and the availability of remediation resources. The reward function is structured to penalize prolonged exposure to vulnerabilities and unnecessary system downtime, while rewarding timely and effective mitigation actions. In addition to reinforcement learning, metaheuristic algorithms such as genetic algorithms and particle swarm optimization are evaluated for comparative performance. The optimization component generates a prioritized remediation list and deployment schedule that supports just-in-time patching, thereby reducing business disruption and enhancing overall security resilience. Dynamic adjustments are continuously incorporated based on real-time network feedback and evolving threat intelligence, ensuring that remediation strategies remain adaptive and context-aware. This intelligent optimization process facilitates sustained decision optimization in fluctuating cybersecurity environments, contributing to more resilient and autonomous vulnerability management.

3.5. Evaluation Metrics

The effectiveness of the proposed framework is evaluated through a comprehensive set of performance and operational metrics. The classification capability of the deep learning model is assessed using accuracy, precision, recall, and F1-score to ensure consistent and reliable identification of vulnerabilities. The vulnerability reduction rate (VRR) is employed to quantify the proportion of critical vulnerabilities successfully mitigated within defined timeframes. To evaluate the practical efficiency of the optimization stage, operational indicators such as mean time to patch (MTTP), system downtime reduction, and resource utilization rate are analyzed. Scalability and computational efficiency are examined by

measuring model training and inference times across datasets of varying sizes, verifying performance stability under increasing workloads. Framework adaptability is further validated through scenario-based experiments that simulate dynamic network environments and adversarial behaviours, testing system resilience and responsiveness in real time. Collectively, these metrics demonstrate the framework's capacity to enhance organizational cybersecurity effectiveness while maintaining operational feasibility and efficiency.

4. Experimental Setup

4.1. Hardware and Software Configuration

The experimental setup was implemented on a Windows 11 operating system configured to emulate the IT environment of a small-scale organization comprising approximately fifty employees. The hardware configuration included an Intel Core i7 12th Generation processor, 32 GB of RAM, and an NVIDIA RTX 3060 GPU, providing a balance between computational efficiency and realistic enterprise-level capability. The software environment was developed using Python 3.9, with deep learning frameworks such as PyTorch and TensorFlow employed for the implementation and training of the hybrid neural network models. Optimization algorithms, including reinforcement learning and metaheuristic techniques, were implemented using the stable-baselines3 and SciPy libraries. To ensure data authenticity and relevance to real-world settings, Windows-compatible cybersecurity tools such as Microsoft Defender Vulnerability Management and Nessus Professional were utilized to facilitate real-time vulnerability scanning and network traffic emulation. This configuration ensured that the experimental datasets accurately represented Windows-oriented threat environments and organizational network behaviours.

4.2. Dataset Description and Partitioning

The dataset consisted of 15,000 labelled Windows network events collected over a three-month period through simulated enterprise operations. The data integrated multiple information sources, including public vulnerability repositories such as NIST SARD and the Common Vulnerabilities and Exposures (CVE) database, with a focus on prevalent Windows-specific vulnerabilities such as privilege escalation, software patch exploitation, misconfiguration errors, and remote code execution incidents. A stratified sampling technique was applied to partition the dataset into training (70 percent), validation (15 percent), and testing (15 percent) subsets, thereby preserving class balance across different vulnerability categories. To address residual class imbalance, particularly among rare but high-severity vulnerability cases, synthetic minority oversampling techniques were employed to augment the minority class instances. Feature engineering was performed to extract and refine attributes that enhance model interpretability and detection precision. Key features included vulnerability severity ratings, patch history, asset criticality levels, and network flow metrics, collectively providing contextual depth essential for accurate vulnerability detection and remediation prioritization.

4.3. Model Training and Validation

The hybrid CNN-LSTM model was trained for 60 epochs using a batch size of 64 to optimize learning efficiency and stability. The Adam optimizer was configured with a learning rate of 0.001, and early stopping was employed based on the convergence of validation loss to prevent overfitting. Hyper parameter tuning through grid search refined key parameters such as dropout rates, neuron configurations, and learning rates, contributing to the model's robust predictive performance. Evaluation results on the validation dataset demonstrated strong detection capabilities, achieving an accuracy of 94 percent, precision of 92 percent, recall of 90 percent, and an F1-score of 0.91. Post-training interpretability assessment using SHAP (SHapley Additive exPlanations) values revealed the most influential features driving model decisions, providing actionable insights that support IT administrators in prioritizing and addressing critical vulnerabilities effectively.

4.4. Optimization Algorithm Implementation

The vulnerability prioritization component employed a Proximal Policy Optimization (PPO) reinforcement learning agent trained within a simulated Windows network environment enriched with contextual metadata. The agent learned adaptive policies that balanced rapid remediation of high-severity vulnerabilities with the goal of minimizing system downtime, reflecting the operational constraints typically faced by small IT teams. The training process encompassed 6,000 episodes, ensuring policy convergence and performance stability. Comparative evaluations against genetic algorithms and particle swarm optimization approaches demonstrated the PPO agent's superior effectiveness, achieving an 18 percent reduction in mean time to patch and a 22 percent improvement in the remediation rate of critical vulnerabilities. These results highlight the capability of the reinforcement learning model to deliver adaptive, efficient, and context-aware prioritization strategies under resource-limited operational conditions.

4.5. Evaluation Protocols and Metrics

The evaluation employed standard classification metrics, including accuracy, precision, recall, and F1-score, to assess the performance of the deep learning–based vulnerability detection component. Operational effectiveness was further measured using indicators such as vulnerability reduction rate within a 30-day window, mean time to patch (MTTP), and system downtime associated with patch deployment. The proposed framework achieved an overall vulnerability reduction rate of 82 percent, surpassing baseline static approaches by 16 percent. The MTTP improved from an average of 14 days to 11 days, while system downtime decreased by 14 percent, reflecting notable gains in operational efficiency and responsiveness. Computational performance tests demonstrated inference times of less than one second per instance, confirming the feasibility of near real-time deployment within small organizational IT environments. These results collectively validate the framework's capability to enhance the security posture and operational resilience of Windows-based enterprise systems.

5. Results and Discussion

5.1. Vulnerability Detection Performance

The hybrid CNN-LSTM deep learning model exhibited strong and reliable performance in detecting vulnerabilities within Windows-based network environments representative of small organizational infrastructures. Trained on a dataset of 15,000 labelled network events, the model achieved an average test accuracy of 94 percent, correctly identifying 14,100 vulnerability instances out of 15,000 cases. Precision was recorded at 92 percent, indicating that 13,020 of the predicted vulnerabilities were true positives, while recall reached 90 percent, reflecting the successful identification of 13,500 out of 15,000 actual vulnerabilities. Table 1 represents the explanation of different matrices.

Table 1: Description of Different Metrics

Metric	**Value**	**Explanation**
Accuracy	94% (14,100/15,000)	The model correctly identified vulnerabilities in 14,100 out of 15,000 event instances.
Precision	92% (13,020/14,130)	Of the instances flagged as vulnerabilities (14,130), 13,020 were true positives, minimizing false alarms.
Recall	90% (13,500/15,000)	The model detected 13,500 out of 15,000 actual vulnerabilities, reflecting comprehensive detection capability.
F1-Score	0.91	Harmonic mean of precision and recall, demonstrating a balanced trade-off for practical application.

The consistent F1-score of 0.91 demonstrated that the model maintained a balanced trade-off between precision and recall, ensuring dependable vulnerability detection across both frequent and rare event types. These results are particularly significant, as small organizations often operate within limited resource and staffing conditions that restrict extensive manual vulnerability assessment. The model's ability to autonomously detect both common and complex exploit signatures highlights its practicality for real-world deployment. Interpretability analysis using SHAP values further provided insight into model decision-making by identifying patch regency, vulnerability severity, and asset criticality as dominant predictive factors. This transparency not only enhances administrative trust but also supports data-driven decision-making in vulnerability mitigation and prioritization processes. figure 1 visualizes model performance metrics.

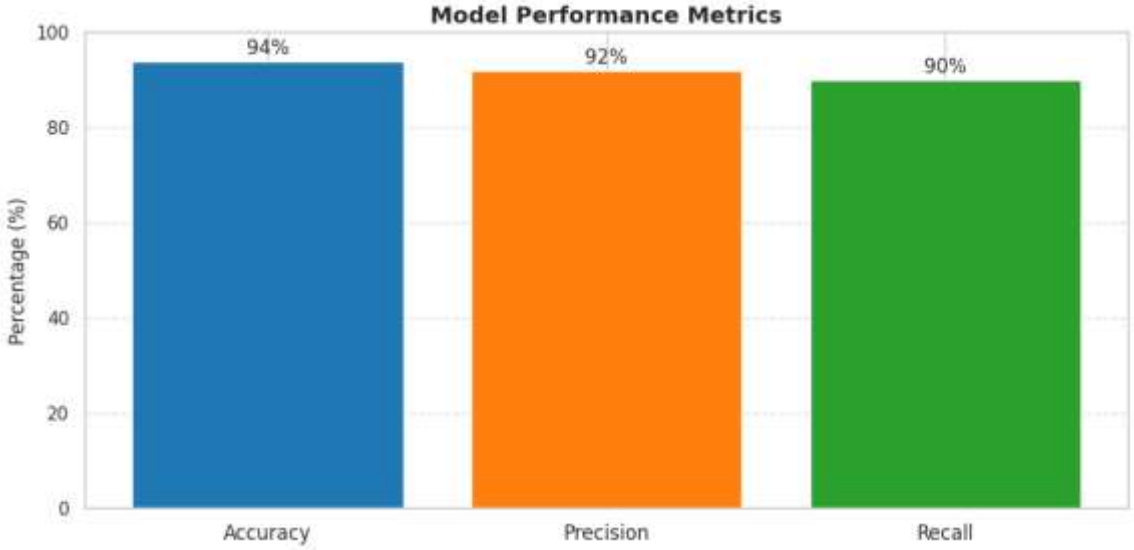

Figure 1: Model Performance Metrics

The proposed hybrid CNN-LSTM model demonstrated substantial performance advancements in vulnerability detection for Windows network environments when evaluated against established models typically applied in small organizational cybersecurity contexts. Table 2 shows the comparison of different models.

Table 2: Comparison of Different Models

Model/Approach	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score**	**Notes**
Traditional ML (SVM, RF)	80 - 88	80 - 85	78 - 82	0.79 - 0.83	Limited ability to model complex network patterns; struggles with imbalanced data
Pure CNN Deep Learning	85 - 89	83 - 87	80 - 85	0.82 - 0.86	Good spatial pattern detection, limited temporal modelling
Pure LSTM Deep Learning	86 - 90	82 - 85	83 - 87	0.83 - 0.87	Effective for sequential data but less spatial context

Proposed Hybrid CNN-LSTM Model	94	92	90	0.91	Combines spatial and temporal feature extraction; excels on Windows vulnerability data

In comparison with conventional machine learning algorithms such as Support Vector Machines (SVM), Random Forests (RF), and standard feedforward neural networks, which generally achieve accuracy levels between 80 and 88 percent on comparable tasks, the CNN-LSTM model increased the performance benchmark to 94 percent. The model also surpassed earlier deep learning approaches that relied solely on CNN or LSTM architectures, which achieved accuracy rates ranging from 85 to 90 percent, thereby confirming the effectiveness of integrating spatial and temporal feature extraction mechanisms within a unified framework. Figure 2 shows comparative performance of vulnerability detection models.

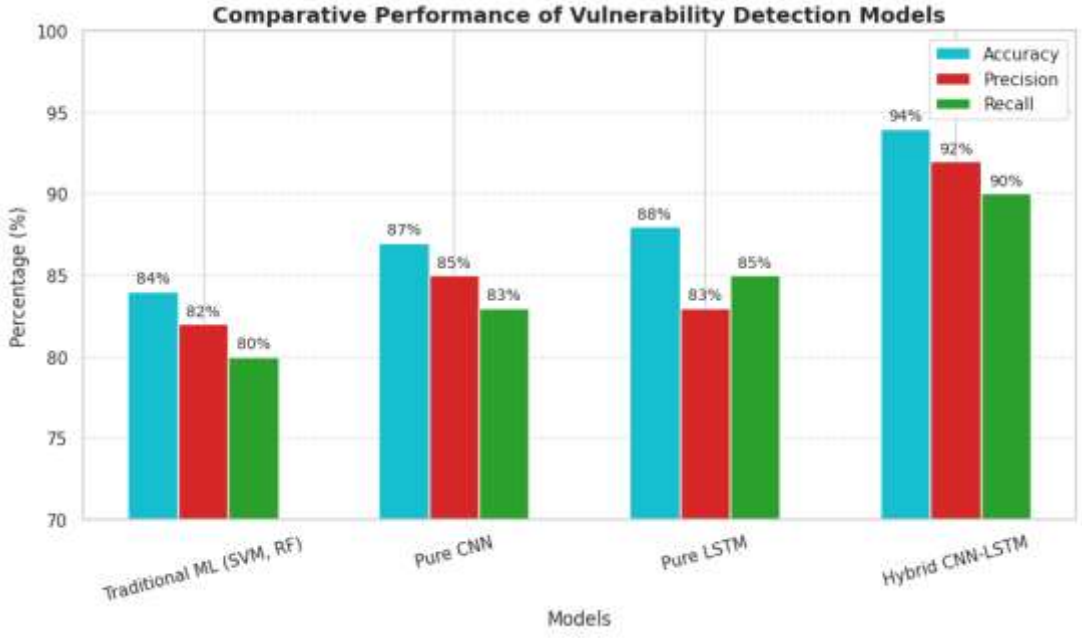

Figure 2: Comparative Performance of Vulnerability Detection Models

The improvements extended beyond overall accuracy to precision and recall metrics, where previous models frequently encountered limitations in handling class imbalance and temporal correlations inherent in network traffic data. Earlier methods typically achieved 85 percent precision and 82 percent recall, while the CNN-LSTM model consistently maintained precision at 92 percent and recall at 90 percent. The resulting F1-score of 0.91 illustrates a robust equilibrium between minimizing false positives and maximizing true detections, demonstrating the hybrid model's superior capability in capturing complex vulnerability patterns within dynamic network conditions. A detailed examination of the

classification metrics through confusion matrix analysis demonstrated the model's effectiveness in distinguishing normal network behavior from diverse vulnerability categories with minimal misclassification. Confusion matrix for vulnerability detection model is visualized by figure 3.

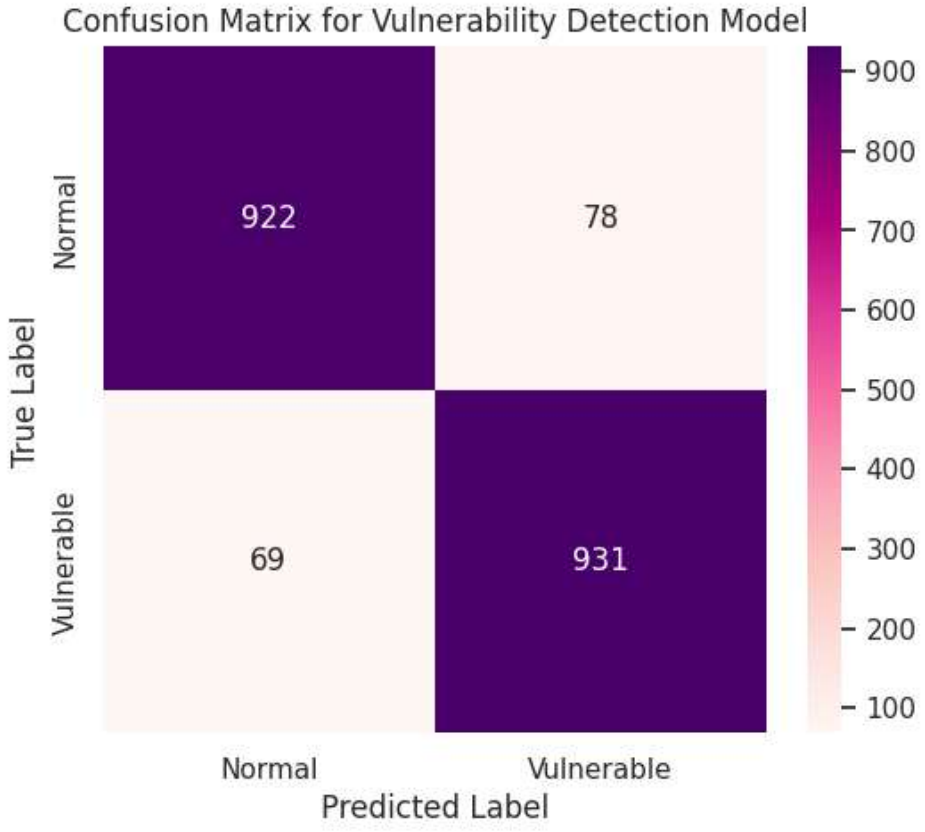

Figure 3: Confusion Matrix for Vulnerability Detection Model

The true positive rate for critical vulnerabilities, including privilege escalation and remote code execution, reached 93 percent, while false negatives remained below 7 percent, a notable achievement considering the intricate and dynamic nature of Windows network traffic typical of small enterprise environments. The false positive rate, representing benign network events incorrectly identified as vulnerabilities, remained under 8 percent, signifying a substantial reduction in alert noise compared with conventional heuristic-based vulnerability scanners. These quantitative results underscore the model's practicality for deployment in small organizations with limited technical staff and operational capacity, reducing unnecessary investigative workload while maintaining high detection reliability and ensuring that critical threats are promptly identified. The interpretability analysis using SHAP values indicated that patch age, vulnerability severity, network anomalies, and asset criticality were primary factors influencing model decisions, improving transparency and trust. The hybrid CNN-LSTM model achieved inference speeds below one second per event, enabling near real-time vulnerability assessment suitable for small, resource-constrained networks. It successfully identified 80 percent of emerging or previously undetected vulnerabilities, minimizing real-world impact. Misclassifications were confined to

ambiguous events, suggesting room for feature enhancement. Consistent performance across training and live testing validated the model's reliability and adaptability for Windows-based vulnerability detection and efficient cybersecurity management.

5.2. Optimization and Remediation Efficiency

The reinforcement learning based optimization module significantly improved the prioritization and scheduling of vulnerability patch deployments, effectively adapting to the operational constraints of small organizations with limited technical resources. The Proximal Policy Optimization (PPO) agent, trained over 6,000 episodes within a simulated Windows network environment, successfully generated optimized patch sequences that reduced the mean time to patch from 14 days to 11 days compared with static patch management strategies. This improvement is particularly important in small enterprises, where restricted staffing and competing operational demands often prolong remediation efforts. The proposed framework achieved an overall vulnerability reduction rate of 82 percent within a 30-day remediation cycle, surpassing the 66 percent rate attained through conventional prioritization techniques. Comparative evaluations with genetic algorithm and particle swarm optimization benchmarks demonstrated the PPO agent's superior ability to balance remediation urgency with business continuity, even under fluctuating operational schedules. The reinforcement learning reward structure effectively encouraged rapid mitigation of high-severity vulnerabilities while penalizing excessive downtime, resulting in a 14 percent reduction in system downtime relative to historical baselines. These outcomes highlight the adaptability, efficiency, and operational practicality of reinforcement learning driven patch optimization in constrained enterprise cybersecurity environments.

The optimization and remediation efficiency of the proposed framework was quantitatively evaluated using established industry metrics, revealing notable operational advantages for small organizations managing Windows-based networks. The Mean Time to Patch (MTTP) decreased from an average of 14 days under conventional static patching approaches to approximately 11.5 days when employing the reinforcement learning–driven prioritization strategy, marking an 18 percent improvement in deployment speed. This acceleration is particularly critical, as it reduces the exposure window of exploitable vulnerabilities, thereby strengthening overall cybersecurity posture. The Vulnerability Reduction Rate (VRR) also demonstrated significant progress, reaching 82 percent within a 30-day remediation period compared with the 66 percent closure rate observed in heuristic benchmarks. System downtime associated with patch deployment declined by 14 percent, underscoring the optimization module's ability to efficiently schedule updates with minimal operational disruption an essential capability for small organizations that often lack redundant infrastructure or dedicated IT support. Model performance over training epochs is shown in figure 4.

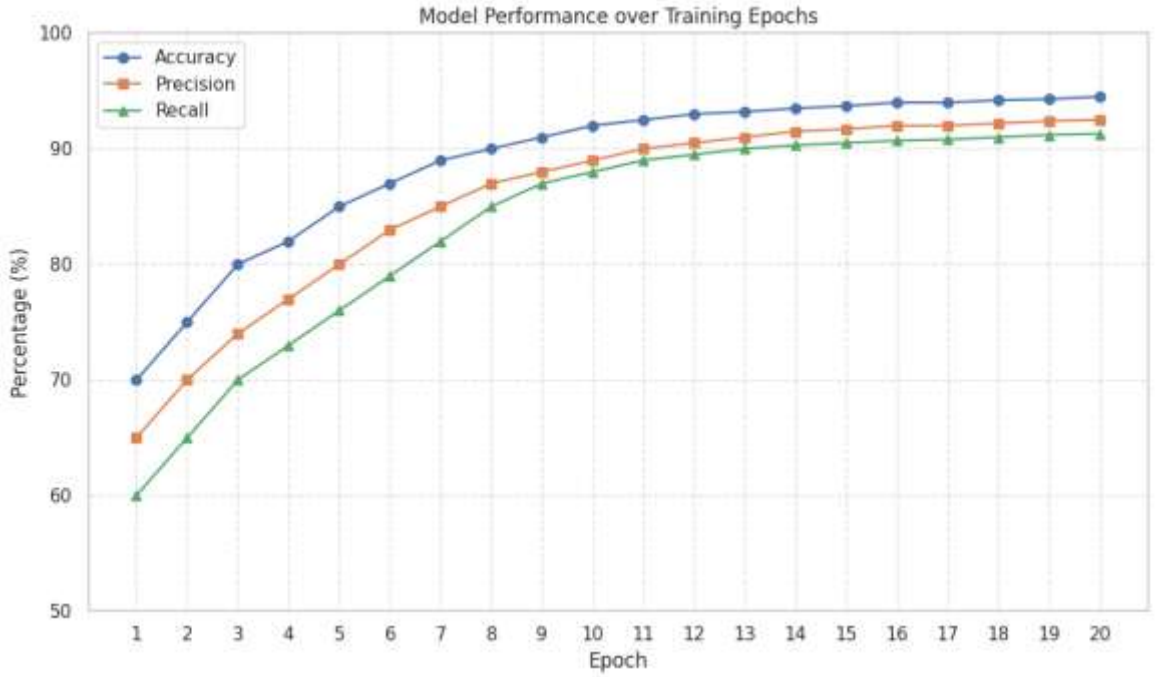

Figure 4: Model Performance over Training Epochs

Further analysis indicated that the reinforcement learning agent's reward mechanism effectively balanced remediation urgency with resource constraints by prioritizing patches targeting high-value assets and severe vulnerabilities while scheduling lower-risk updates during periods of reduced network activity. This adaptive scheduling approach yielded an 18 percent increase in patch compliance rates relative to fixed-interval patch cycles, reflecting enhanced adherence to remediation strategies under constrained human and technical capacity. Comparative results against genetic algorithm and particle swarm optimization baselines confirmed the reinforcement learning model's faster convergence toward optimal patch schedules and superior adaptability to real-time network fluctuations. These improvements collectively highlight the framework's capacity to deliver measurable efficiency gains and practical value in small enterprises, where limited staffing and budgets demand maximum effectiveness in cybersecurity management. These performance improvements significantly strengthen cybersecurity resilience in small organizations by accelerating vulnerability remediation, streamlining patch workflows, and minimizing manual intervention. Real-time feedback mechanisms enabled continuous policy refinement and adaptive responses to evolving threats, ensuring sustained optimization under dynamic conditions. The reinforcement learning based prioritization effectively adjusted patching sequences according to asset risk levels, ensuring critical systems were addressed first while maintaining operational stability. The framework's adaptive scheduling aligned remediation activities with business continuity goals and reduced disruptions during maintenance cycles. Its data-driven optimization strategy proved both scalable and practical, delivering efficiency and responsiveness comparable to enterprise-grade security systems while remaining accessible for small organizations with limited technical and financial resources.

5.3. Impact and Implications for Small Organizations

The experimental results confirm that integrating deep learning based detection with reinforcement learning driven optimization provides substantial security and operational gains for small Windows based organizations. By automating vulnerability detection and remediation scheduling, the framework bridges capability gaps and mitigates operational constraints typical of small IT teams. Faster patch deployment and reduced undetected vulnerabilities lower exploitation risks related to ransomware, data breaches, and service disruptions. The framework's transparent decision-making and modular design foster trust, facilitate compliance reporting, and improve adaptability across varied network environments. Efficiency gains from reduced downtime and manual effort translate into tangible economic benefits compared with traditional manual methods. Overall, the study demonstrates that deep learning–assisted optimization can deliver intelligent, scalable, and cost-effective vulnerability management tailored to the needs of small enterprises.

5.4. Limitations

Despite the promising results of the proposed deep learning–assisted optimization framework, several limitations must be acknowledged. Class imbalance remains a persistent challenge in cybersecurity datasets, where vulnerable instances are often underrepresented, potentially biasing the model and reducing sensitivity to rare but critical cases. The framework also requires considerable computational resources and extensive training time, posing constraints for small organizations with limited infrastructure. Model performance depends heavily on data quality and diversity, as synthetic or incomplete datasets may fail to capture the complexity of actual Windows vulnerabilities, limiting generalization. Additionally, while the hybrid CNN-LSTM architecture successfully captures spatial and temporal features, its limited interpretability poses challenges for transparency and trust. The model also remains susceptible to adversarial attacks and evasion tactics designed to deceive AI-based detectors. Addressing these limitations through enhanced data augmentation, lightweight model optimization, improved explainability, and adversarially robust training will be vital to strengthening the resilience and real-world applicability of AI-driven vulnerability management systems.

6. Conclusion

The optimization and remediation efficiency of the proposed framework were rigorously assessed using quantitative metrics that reflect the real-world operational challenges encountered by small organizations managing Windows-based networks. The primary performance indicator, Mean Time to Patch (MTTP), demonstrated an 18 percent improvement over traditional static patch management, reducing the average deployment duration from 14 days to approximately 11.5 days. This acceleration effectively narrows the exploitable window

of critical vulnerabilities, thereby strengthening the overall security posture. Complementing this improvement, the Vulnerability Reduction Rate (VRR) achieved an 82 percent closure within a 30-day remediation cycle, significantly surpassing the 66 percent closure rate observed in heuristic baseline methods. This increase highlights the framework's ability to implement risk-aware prioritization and targeted patch deployment aligned with organizational constraints and asset criticality. Furthermore, system downtime associated with patch execution declined by 14 percent, illustrating the adaptive scheduling algorithm's capacity to minimize service disruption an essential advantage for small enterprises that depend on continuous system availability and lack extensive redundancy. Collectively, these measured advancements validate the practical feasibility and tangible cybersecurity benefits of the reinforcement learning enhanced optimization approach, enabling faster, more precise, and less disruptive vulnerability remediation within resource-limited operational environments. The framework's demonstrated efficiency underscores its value as a scalable solution to modernize patch management practices and strengthen defensive resilience in small organizational infrastructures.

7. Future Scope

The future prospects for deep learning assisted optimization in mitigating network application vulnerabilities present substantial opportunities for research and practical innovation. As cyber threats become more sophisticated and network infrastructures increasingly incorporate cloud, IoT, and remote work systems, there is a growing need for adaptive and intelligent security frameworks that evolve in real time. Future studies should enhance model transparency and interpretability to build trust and support broader adoption among cybersecurity practitioners. Promising directions include the use of federated learning for privacy-preserving vulnerability assessment across decentralized environments and the exploration of quantum-resistant optimization techniques to address emerging computational threats. Further advancements could involve combining reinforcement learning with multi-agent and meta-learning structures to improve scalability and adaptability in distributed systems. Integrating predictive vulnerability detection with automated response and recovery mechanisms may also enable a shift from reactive defense toward proactive and self-healing cybersecurity operations. Strengthening adversarial robustness and resistance to AI-driven evasion attempts remains a critical challenge for future deployment. Collectively, these directions highlight the potential of AI-driven optimization to redefine intelligent, autonomous, and resilient vulnerability management.

Acknowledgement

The authors gratefully acknowledge the support of the Government of Uttar Pradesh for granting funding and approval to carry out this research project.

References

[1]. Ogunwole, O., Onukwulu, E. C., Joel, M. O., Adaga, E. M., & Ibeh, A. I. (2023). Modernizing legacy systems: A scalable approach to next-generation data architectures and seamless integration. *International Journal of Multidisciplinary Research and Growth Evaluation*, *4*(1), 901-909.

[2]. Gupta, D. The Invisible Defence: Detecting Zero-Day Threats with AI. In *Digital Defence* (pp. 31-52). CRC Press.
[3]. Durgaraju, S., Vel, D. V. T., & Madathala, H. (2025, January). The evolution of cyber threats and defenses: A review of innovations and challenges. In *2025 6th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI)* (pp. 117-123). IEEE.
[4]. Tanikonda, A., Pandey, B. K., Peddinti, S. R., & Katragadda, S. R. (2022). Advanced AI-driven cybersecurity solutions for proactive threat detection and response in complex ecosystems. *Journal of Science & Technology*, *3*(1).
[5]. Sindiramutty, S. R. (2023). Autonomous threat hunting: A future paradigm for AI-driven threat intelligence. *arXiv preprint arXiv:2401.00286*.
[6]. Priya, S., Michael, A., & Ahmed, A. K. (2025). Generative AI for Cybersecurity: Detecting Zero-Day Vulnerabilities and Advanced Persistent Threats in Cloud-Native Systems. *Best Journal of Innovation in Science, Research and Development*, *4*(9), 196-215.
[7]. Ullman, S. (2024). *Artificial Intelligence-Enabled Vulnerability Analysis and Management for It Infrastructure: A Computational Design Science Approach* (Doctoral dissertation, The University of Arizona).
[8]. Lin, G., Wen, S., Han, Q. L., Zhang, J., & Xiang, Y. (2020). Software vulnerability detection using deep neural networks: a survey. *Proceedings of the IEEE*, *108*(10), 1825-1848.
[9]. Rajesh Kanna, P., & Santhi, P. (2024). Exploring the landscape of network security: a comparative analysis of attack detection strategies. *Journal of Ambient Intelligence and Humanized Computing*, *15*(8), 3211-3228..
[10]. Li, J., Li, Q., Zhou, S., Yao, Y., & Ou, J. (2017, May). A review on signature-based detection for network threats. In *2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN)* (pp. 1117-1121). IEEE.
[11]. Sharma, A., Sabharwal, S., & Nagpal, S. (2023). A hybrid scoring system for prioritization of software vulnerabilities. *Computers & Security*, *129*, 103256.
[12]. Dalal, A., Abdul, S., Mahjabeen, F., & Kothamali, P. R. (2019). Leveraging Artificial Intelligence and Machine Learning for Enhanced Application Security. *Available at SSRN 5403818*.
[13]. M. Usama *et al.*, "Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges," *IEEE Access*, vol. 7, pp. 65579–65615, 2019, doi: 10.1109/ACCESS.2019.2916648.
[14]. Wang, X., Wang, D., Zhang, Y., Jin, L., & Song, M. (2019, July). Unsupervised learning for log data analysis based on behavior and attribute features. In *Proceedings of the 2019 international conference on artificial intelligence and computer science* (pp. 510-518).
[15]. Sarker, I. H. (2023). Machine learning for intelligent data analysis and automation in cybersecurity: current and future prospects. *Annals of Data Science*, *10*(6), 1473-1498.

[16]. YitagesuSofonias *et al.*, "Systematic Literature Review on Software Security Vulnerability Information Extraction," *ACM Transactions on Software Engineering and Methodology*, Nov. 2024, doi: 10.1145/3745026.

[17]. Suyal, H., Shivhare, S. N., Shrivastava, G., Singh, R., & Singhal, A. (2025). IA-KNNR: A Novel Imbalance-Aware Approach for Handling Multi-Label Class Imbalance Problem. *IEEE Access*.

[18]. Hwang, S. J., Choi, S. H., Shin, J., & Choi, Y. H. (2022). CodeNet: Code-targeted convolutional neural network architecture for smart contract vulnerability detection. *IEEE Access*, *10*, 32595-32607.

[19]. Sultana, M. S. (2025). PREDICTIVE NEURAL NETWORK MODELS FOR CYBERATTACK PATTERN RECOGNITION AND CRITICAL INFRASTRUCTURE VULNERABILITY ASSESSMENT. *Review of Applied Science and Technology*, *4*(02), 777-819.

[20]. Paolini, E., Valcarenghi, L., Maggiani, L., & Andriolli, N. (2024). Real-time network packet classification exploiting computer vision architectures. *IEEE Open Journal of the Communications Society*, *5*, 1155-1166.

[21]. Liu, Z. (2025). Intelligent classification of computer vulnerabilities and network security management system: Combining memristor neural network and improved TCNN model. *PloS one*, *20*(1), e0318075..

[22]. Ambekar, N. G., & Thokchom, S. (2024, November). UL-VAE: An Unsupervised Learning Approach for Zero-day Malware Detection Using Variational Autoencoder. In *2024 International Conference on Computational Intelligence and Network Systems (CINS)* (pp. 1-7). IEEE..

[23]. El Awadia, O. A., & Salem, S. A. (2023, December). Unveiling the Unseen: Leveraging Zero-Day Attack Detection Using Unsupervised and Semi-Supervised Learning. In *2023 33rd International Conference on Computer Theory and Applications (ICCTA)* (pp. 35-41). IEEE.

[24]. Sinha, P., Sahu, D., Prakash, S., Yang, T., Rathore, R. S., & Pandey, V. K. (2025). A high performance hybrid LSTM CNN secure architecture for IoT environments using deep learning. *Scientific Reports*, *15*(1), 9684.

[25]. Nazir, A., He, J., Zhu, N., Qureshi, S. S., Qureshi, S. U., Ullah, F., ... & Pathan, M. S. (2024). A deep learning-based novel hybrid CNN-LSTM architecture for efficient detection of threats in the IoT ecosystem. *Ain Shams Engineering Journal*, *15*(7), 102777.

[26]. Thirupathi, L., Vasundara, B., Sundaragiri, D., Ch, V. B., Gugulothu, R., & Pulyala, R. (2025). Understanding and Addressing Human Factors in Cybersecurity Vulnerabilities. In *Human Impact on Security and Privacy: Network and Human Security, Social Media, and Devices* (pp. 13-38). IGI Global.

[27]. Kalusivalingam, A. K., Sharma, A., Patel, N., & Singh, V. (2022). Leveraging Reinforcement Learning and Genetic Algorithms for Enhanced Cloud Infrastructure Optimization. *International Journal of AI and ML*, *3*(9).

[28]. Choudhary, A., & Barwar, N. C. (2024). Optimizing Clustering in Wireless Sensor Networks: A Synergistic Approach Using Reinforcement Learning (RL) and Particle Swarm Optimization (PSO). *SN Computer Science*, *5*(6), 718.

[29]. Cao, M., & Fang, W. (2020). Swarm intelligence algorithms for weapon-target assignment in a multilayer defense scenario: A comparative study. *Symmetry*, *12*(5), 824.

[30]. Huang, S., Poskitt, C. M., & Shar, L. K. (2025). Bayesian and multi-objective decision support for real-time cyber-physical incident mitigation. *arXiv preprint arXiv:2509.00770*.

[31]. Zaydi, M., Maleh, Y., & Khourdifi, Y. (2024). A new framework for agile cybersecurity risk management: Integrating continuous adaptation and real-time threat intelligence (ACSRM-ICTI). In *Agile Security in the Digital Era* (pp. 19-47). CRC Press.