International Journal of Applied Mathematics
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

AUTONOMOUS POLICY DRIFT DETECTION IN CDN-WAF PIPELINES USING GRAPH NEURAL NETWORKS

Vinod Kumar Thottempudi
New England College and Henniker, New Hampshire
vinod.thottempudi@gmail.com

Abstract

The increasing complexity of web-based attacks and the dynamic nature of web traffic trends are placing significant burdens on the effectiveness of security policy provisions within Content Delivery Network (CDN) and Web Application Firewall (WAF) pipelines. As adversarial strategies continue to evolve, traditionally fixed sets of detection rules are becoming obsolete. This leads to policy drift, wherein existing detection systems lose their efficacy. This review paper aims to analyze the application of autonomous drift detection using Graph Neural Networks (GNNs) and reinforcement learning-based frameworks as a solution to this challenge. It explores the relevance of adaptive control systems, actor-critic architectures, anomaly detection techniques, and governance-aware AI deployments, particularly in the context of their integration into CDN-WAF infrastructures. The paper highlights how GNNs can capture intricate structural patterns within web traffic data, how continuous security policies can be dynamically learned using deep reinforcement learning, and how responsible AI practices can ensure compliance and transparency. Furthermore, it provides an overview of current trends, defines key architectural components, and discusses the challenges associated with deploying autonomous mechanisms for detecting policy drift in high-security environments.

Keywords Policy drift, Graph Neural Networks, Web Application Firewall, Reinforcement learning

1. Introduction

Web Application Firewalls (WAFs) and Content Delivery Networks (CDNs) are considered essential components of modern web infrastructure. While CDNs focus on optimizing the delivery of digital content by caching data closer to end-users, WAFs are responsible for scanning and filtering incoming HTTP traffic to prevent malicious access. A combined CDN-WAF pipeline is typically employed to ensure both high performance and secure service delivery. However, policy drift remains a persistent challenge in the face of rapidly evolving

Received: Aug 11, 2025

Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
threat environments and dynamic user behavior. Policy drift occurs when implemented rules or detection strategies gradually become ineffective due to changes in data patterns, adversarial behavior, or internal modifications within the pipeline.

The emergence of autonomous systems and Graph Neural Networks (GNNs) introduces a novel approach to detecting and responding to such drifts. The strength of GNNs lies in their ability to model and learn from the relational and temporal characteristics of network events and threat signatures. Moreover, the integration of reinforcement learning, dynamic feature weighting, and actor-critic architectures enhances the resilience of autonomous detection models against policy drift. This paper presents a state-of-the-art review of autonomous drift detection in CDN-WAF pipelines, with particular emphasis on the application of GNNs, reinforcement learning, and emerging risk-aware architectural frameworks.

2. Policy Drift in Dynamic Threat Environments

The increasing complexity and sophistication of cyber-attacks demand robust and adaptive security responses. Traditional rule-based intrusion detection systems tend to become obsolete over time, as attackers continuously evolve their techniques, resulting in previously unseen threats or increased false positives. In response to these challenges, autonomous and adaptive drift detection schemes have emerged to ensure the sustained effectiveness of security policies.

This challenge, commonly referred to as policy drift, has been addressed through the application of actor–critic learning models that adjust detection patterns in real time. In such architectures, actor networks are responsible for developing adaptive policies, while critic networks evaluate their performance based on real-time feedback. Feature reweighting is a critical requirement in this context, as not all features remain equally informative for threat detection when drift occurs. One such approach is DriftShield, an autonomous framework for monitoring feature importance and dynamically reconfiguring models in highly dynamic data environments [1].

Similarly, incremental learning methods have been applied to detect feature drift in intrusion detection systems. These approaches often employ reinforcement learning–based voting schemes, which consist of modular decision nodes that learn to maintain stability across features over time. The system architecture adapts in response to observed changes, leveraging historical data to construct updated classification models. This approach ensures that shifts in data distributions are accommodated through revised policies without requiring complete retraining, thereby maintaining system responsiveness and operational efficiency [2].

Received: Aug 11, 2025

International Journal of Applied Mathematics
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

3. Autonomous Drift Detection and Behavior Modeling

This concept has been exemplified by the development of autonomous decision-making in high-risk environments such as autonomous racing, which offers valuable insights applicable to cybersecurity. In such racing environments, adaptive control systems are designed to respond in real-time to environmental signals, road conditions, and internal vehicle dynamics. These systems rely on state–action feedback models and data-driven training to generate optimal navigation strategies based on learned experience and real-time sensor input [3] [14] .

The logic applied in autonomous driving—such as detecting driver intention—is analogous to cybersecurity applications, where the objective is to detect user or attacker intent. Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) models are powerful machine learning tools capable of capturing latent behavioral sequence patterns, enabling the prediction of anomalous or atypical modes of operation. Similarly, the emergence of abnormal HTTP request patterns or unexpected sequences within a session may signal the onset of drift in CDN-WAF pipelines, prompting the need for adaptive policy modification to maintain system integrity and detection accuracy [4] [18].

Figure 1 below illustrates the conceptual alignment between autonomous vehicle drift adaptation and CDN-WAF policy drift detection using GNNs. The sequence of external stimuli, internal feature monitoring, and policy adaptation mirrors processes in both domains.

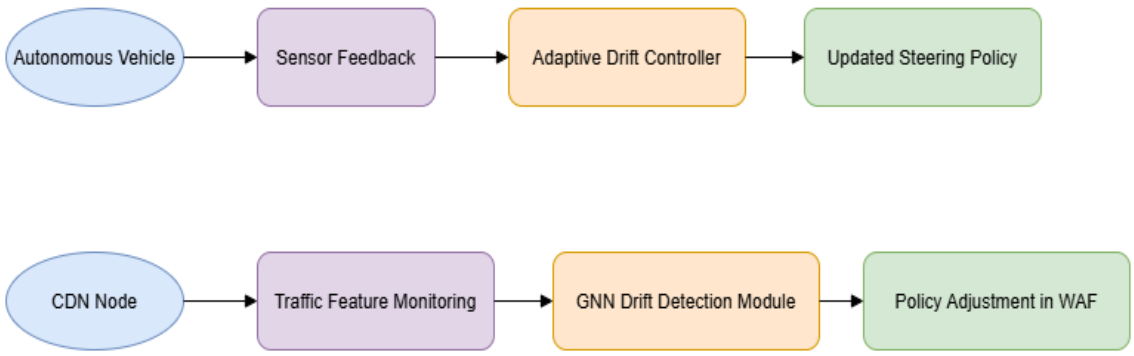

Figure 1: Conceptual comparison between autonomous vehicle drift detection and CDN-WAF policy drift adaptation via GNNs

Source: Adapted from [3], [4]

4. Integration of Deep Reinforcement Learning for Drift Handling

The application of deep reinforcement learning (DRL) to enable decision-making models to adapt dynamically to ongoing model drift is particularly promising. The emergence of Deep

Received: Aug 11, 2025

International Journal of Applied Mathematics
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
Reinforcement Malware Detection (DRMD) demonstrates how agents trained using DRL can ensure that detection policies continuously adapt to concept drift in malware samples. The DRMD framework employs online learning to adjust model parameters when new malware types or variants are encountered, thereby ensuring that the detection model remains effective across an evolving threat landscape [5] [20].

A similar architectural approach can be extended to CDN–WAF pipelines. Rather than analyzing malware binaries, the model would process real-time HTTP traffic, user-agent behavior, and content delivery anomalies. Policy updates could be performed using Q-learning or policy-gradient–based methods, allowing the system to continuously refine its threat classification and rule-matching strategies.

However, DRL-based approaches are susceptible to overfitting short-term transient drifts or noise present in streaming data. To mitigate this risk, memory accumulation mechanisms or ensemble-based strategies can be incorporated to preserve historical decision baselines. Such mechanisms help prevent undesirable oscillatory behavior in policy updates and ensure stability in long-term security policy formation.

5. Security Implications of Autonomy in Drift-Responsive Systems

Adaptive learning agents that are self-directed can function both as defensive tools and as potential vectors for offensive threats. As the autonomy of systems tasked with detecting and countering policy drift increases, so too does the size of their attack surface. Adversarial manipulation becomes a significant concern, particularly for large model-based agents such as those built on deep learning or transformer architectures. These agents can be deceived into misclassifying threats or into relaxing security rules based on the false assumption that an observed drift represents benign user behavior.

The broader overview of security risks introduced by autonomous systems highlights their increased susceptibility due to the complexity of their models and the difficulty in defining strict boundaries for decision-making. In the context of CDN–WAF systems, for instance, an autonomous drift detector may mistakenly interpret a sudden surge in botnet traffic as a natural fluctuation in user behavior, leading to a reduction in security policy strictness. Ensuring robustness, explainability, and incorporating external verification mechanisms are essential to preventing such misjudgments and maintaining reliable decision-making processes [6] [19].

Table 1 summarizes key architectural characteristics of current autonomous drift detection approaches and their associated security implications.

International Journal of Applied Mathematics
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Table 1: Comparison of Autonomous Drift Detection Architectures and Security Considerations

Architecture	Learning Model	Drift Handling Technique	Security Risk	Mitigation Strategy
DriftShield [1]	Actor-Critic RL	Feature Reweighting	Overfitting to short-term drift	Dynamic feature validation
RL-Voting IDS [2]	Incremental RL	Ensemble-based feature tracking	Model staleness	Historical pattern integration
Auto Drifting Vehicles [3]	Control-based RL	Sensor feedback adjustment	Unintended system bias	Continuous retraining
Driver Intention Systems [4]	Supervised ML	Sequential inference	Pattern misclassification	Multi-sensor validation
DRMD [5]	Deep Q-Networks	Online malware adaptation	Adversarial input poisoning	Noise-resistant encoding
Large Agents Security Survey [6]	Multi-agent RL	Architecture risk analysis	Unexplained behaviour	Governance layer insertion

6. Graph Neural Networks in CDN-WAF Drift Detection

Graph Neural Networks (GNNs) offer a natural framework for modeling the complex relationships between entities within CDN–WAF ecosystems. These entities include IP addresses, session tokens, user-agent strings, CDN edge nodes, and firewall rules. GNNs are capable of capturing the structural and contextual integrity of these relationships through embeddings that represent the interconnected entities. By enabling the flow of information across graph nodes, GNNs allow the model to detect anomalous behavior, such as repeated request patterns or the formation of coordinated attack paths.

A common policy used in CDN environments is the blocking of repeated connections from suspicious IP addresses. However, attackers often circumvent this by rotating IPs or using distributed proxy networks. GNN-based models can detect such coordinated anomalies by identifying clusters of seemingly distinct IPs that exhibit similar behavioral patterns and access

Received: Aug 11, 2025

International Journal of Applied Mathematics
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
similar endpoints. It is critical for GNNs to generalize such patterns across the graph structure to effectively respond to policy drift.

In a similar fashion, GNNs can be applied to monitor the long-term effectiveness of WAF rules by linking request payloads with triggered rules and corresponding system responses. Over time, the GNN can identify which graph paths are associated with undetected or misclassified threats—particularly when a rule becomes less effective due to emerging obfuscation techniques or other adversarial tactics. Policy updates can then be prioritized based on rules that show a strong correlation with drift-related misclassifications.

This capability for relational learning and dynamic topology representation gives GNNs a significant advantage over classical detection models, which rely on static features and single-instance evaluations. However, the effectiveness of GNNs is highly dependent on the timely ingestion and processing of updates within the graph. For this reason, integrating GNNs with real-time data streams is essential to maintain accuracy and responsiveness in dynamic threat environments.

7. Risk-Aware and Security-by-Design Architectures

The shift toward autonomous drift detection necessitates the development of non-adaptive, risk-aware architectures. Ensuring the safe operation of AI systems is paramount—especially as they are granted increasing autonomy in decision-making—particularly in highly sensitive environments such as CDN–WAF pipelines. The risks introduced by policy drift can be mitigated by designing system architectures that adhere to security-by-design principles at the model level.

One emerging solution is the implementation of risk-aware AI governance frameworks, which incorporate formal verification, runtime monitoring, and structured model deployment strategies. These frameworks support assurance throughout the AI lifecycle, encompassing phases such as data collection, model training, and post-deployment drift monitoring. Notably, recent research highlights the incorporation of probabilistic risk assessments and adversarial resilience testing as intrinsic elements of these architectures. This approach ensures that fallback mechanisms and external auditing systems are embedded within the drift detection loop, which is particularly critical in the context of CDN–WAF systems [7] [15].

In this regard, adopting a multi-agent perspective offers additional advantages for detecting and managing policy drift. Multi-agent systems can maintain consistency and detect large-scale anomalies by allowing agents to learn autonomously and coordinate their observations across

International Journal of Applied Mathematics
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
distributed network endpoints. Individual agents may specialize in monitoring specific aspects such as CDN edge locations, request frequency, session continuity, or rule-matching within the CDN–WAF pipeline. Consensus mechanisms can be employed to justify the decisions of individual agents, especially when an agent detects drift based on corroboration from peer agents. This collective validation helps reduce false alerts and mitigates the risk of regressive policy adaptation [8] [17].

8. Application of Large Language Models in Policy Drift Detection

Large Language Models (LLMs) have offered significant opportunities for the research of autonomous software management and policy recommendation. The use of LLMs has also proven effective in developing, certifying, and streamlining security configurations within CI/CD and DevOps environments. These models are automatically used to detect policy drift in WAFs, which maintain configuration files and detection rules in structured or semi-structured formats, such as JSON, YAML, or XML.

Enhanced DevOps AI pipelines are capable of making independent adjustments to WAF settings based on prior experiences and newly observed threat models. LLMs can also analyze logs, identify inconsistencies in rule implementations, and suggest policy modifications aligned with best practices. Such systems function as policy evolution engines within the CDN–WAF environment, responding to detected drift events with case-specific remedies. This automation reduces human-induced delays in responding to drift and helps ensure that protection rules remain up to date [9] [13].

However, policies generated by LLMs can often be difficult to interpret. Therefore, audited configurations must remain compliant with enterprise policies. This can be ensured by incorporating transparency layers, version control, and feedback mechanisms to verify proposed changes before implementation.

9. AI-Augmented WAF Systems and CDN Optimization

The use of AI in anomaly detection is proving to be a best practice in modern cybersecurity architecture for optimising the functionality of WAFs. Latency, false positives, and throughput in a CDN–WAF pipeline need to be balanced against overall security effectiveness. AI-based dynamic models allow for WAF policies and signature thresholds to be adjusted in response to changing network traffic patterns, behavioural baselines, and historical attack vectors.

International Journal of Applied Mathematics
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Recent investigations have shown that zero-day attacks can be identified with the help of AI-based Web Application Firewalls, while also reducing performance bottlenecks in high-load scenarios. Anomaly detection modules facilitate local decision-making by integrating detection components with CDN nodes, thereby helping to offload the central WAF and minimise potential bottlenecks. These systems often use unsupervised learning techniques to distinguish between benign anomalies—such as legitimate traffic surges—and actual attack patterns. These processes are implemented in incremental steps, reducing the need for manual intervention and ensuring that, in the event of drift, fixed rules do not misclassify acceptable user behaviour as malicious [10] [16] .

Figure 2 below illustrates how detection accuracy in AI-based WAF systems improves over time as the system adapts to drift using reinforcement learning and graph-based analysis.

Figure 2: Detection Accuracy Improvement Over Time with Autonomous Drift Adaptation in GNN-WAF Systems

Source: Adapted from [10]

10. Governance, Compliance, and Responsible AI in Drift Detection Systems

In addition to technical efficacy, autonomous drift detection systems must be aligned with broader governance and compliance frameworks. Responsible AI principles should be incorporated when deploying autonomous systems that can independently modify security postures. These principles include requirements for explainability, auditability, and compliance with regulatory frameworks such as GDPR and NIST cybersecurity standards.

Received: Aug 11, 2025

International Journal of Applied Mathematics
Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
A quantitative governance mechanism focused on the operationalisation of responsible AI is grounded in metric-based considerations of fairness, robustness, and consistency with ethical principles. Within a CDN–WAF pipeline, this can involve real-time reporting of policy changes, validation logs, and impact measurements for every autonomous decision made by the drift detection engine. Such mechanisms are employed to ensure that increasing autonomy does not compromise accountability [11] [12].

Responsible AI implementations also require a human-in-the-loop approach. Although policy changes may be suggested and even enacted by autonomous systems, security analysts may retain final decision-making authority in high-risk scenarios. This level of controlled autonomy ensures that human judgment remains integral to decisions that affect service availability and user access.

11. Conclusion and Future Directions

Despite the numerous advantages in the field of drift detection, GNNs and reinforcement learning present certain challenges. To begin with, real-time computation is costly with respect to graph updates in large-scale networks. One of the key strands of research focuses on the ability to efficiently utilise large graphs and memory resources without increasing detection latency.

Second, adversarial attacks in which malicious actors deliberately inject patterns to confuse drift detection mechanisms pose an increasingly serious threat. To counter this, systems must be trained using adversarial techniques and evaluated for robustness when exposed to misleading or manipulated inputs.

Third, cross-domain generalisation remains limited. A drift detection model trained in one CDN–WAF environment may not perform effectively in another due to variations in traffic composition, user behaviour, or deployment configurations. Addressing this limitation requires advancements in transfer learning and domain adaptation techniques to improve model portability and applicability.

Finally, ethical and legal concerns are also raised by autonomous systems that filter access to data or services. Key challenges include defining liability in cases of incorrect drift adaptation and ensuring user privacy during traffic inspection processes.

References

Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
[1] Leng, J., You, W., & Yu, L. (2024). A state-of-the-art review on vehicle drift dynamics and autonomous drift control. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 09544070251342056.

[2] Yu, K., Fu, M., Tian, X., Yang, S., & Yang, Y. (2024). Curriculum reinforcement learning-based drifting along a general path for autonomous vehicles. Robotica, 42(10), 3263-3280.

[3] Hossain, M. S. A., Ahammed, A. S., Biswas, D. P., & Obermaisser, R. (2024, September). Impact analysis of data drift towards the development of safety-critical automotive system. In 2024 International Symposium ELMAR (pp. 325-330). IEEE.

[4] Fang, Z., & Zdun, U. (2024, October). Detecting Environment Drift in Reinforcement Learning Using a Gaussian Process. In 2024 IEEE 36th International Conference on Tools with Artificial Intelligence (ICTAI) (pp. 992-999). IEEE.

[5] Muonagor, C., Bensalem, M., & Jukan, A. (2024). Predictive intent maintenance with intent drift detection in next generation network. arXiv preprint arXiv:2404.15091.

[6] Mridul, M. H., Khan, M. F., Rizvee, R. A., & Khan, M. M. (2024). Adaptive Opponent Policy Detection in Multi-Agent MDPs: Real-Time Strategy Switch Identification Using Running Error Estimation. arXiv preprint arXiv:2406.06500.

[7] Muonagor, C., Bensalem, M., & Jukan, A. (2024, November). Performance Analysis of Learning-based Intent Drift Detection Algorithms in Next Generation Networks. In 2024 IEEE Latin-American Conference on Communications (LATINCOM) (pp. 1-6). IEEE.

[8] Kalenberg, K., Müller, H., Polonelli, T., Schiaffino, A., Niculescu, V., Cioflan, C., ... & Benini, L. (2024). Stargate: Multimodal sensor fusion for autonomous navigation on miniaturized uavs. IEEE Internet of Things Journal, 11(12), 21372-21390.

[9] Rauba, P., Seedat, N., Kacprzyk, K., & van der Schaar, M. (2024). Self-healing machine learning: A framework for autonomous adaptation in real-world environments. Advances in Neural Information Processing Systems, 37, 42225-42267.

[10] Yu, X., Ruan, L., Evans, J. S., & Wong, E. (2024, June). Novel concept drift detection and adaptation (CDDA) framework for human-to-machine (H2M) applications over future communication networks. In ICC 2024-IEEE International Conference on Communications (pp. 5509-5514). IEEE.

Received: Aug 11, 2025

Volume 38 No. 11s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
[11] Joshi, N. (2024). Optimizing Real-Time ETL Pipelines Using Machine Learning Techniques. Available at SSRN 5054767.

[12] Chen, J., Dai, S., Chen, F., Lv, Z., Tang, J., & Han, L. (2024, October). Edge-cloud collaborative motion planning for autonomous driving with large language models. In 2024 IEEE 24th International Conference on Communication Technology (ICCT) (pp. 185-190). IEEE.

[13] Zhao, S., Zhang, J., He, X., He, C., Hou, X., Huang, H., & Han, J. (2024). A Harmonized approach: Beyond-the-limit control for autonomous vehicles balancing performance and safety in unpredictable environments. IEEE Transactions on Intelligent Transportation Systems, 25(11), 15827-15840.

[14] Gudepu, V., Chintapalli, V. R., Castoldi, P., Valcarenghi, L., Tamma, B. R., & Kondepu, K. (2024). The drift handling framework for open radio access networks: An experimental evaluation. Computer Networks, 243, 110290.

[15] Park, T., Ki, J., & Ho, J. Y. (2024). Sectoral innovation policy adopting socio-technical systems (STS) perspectives: The case of autonomous vehicles (AVs) in five leading countries. Technological Forecasting and Social Change, 209, 123758.

[16] Thota, P. K. (2024). A Generative AI Framework for Autonomous Infrastructure Management in Cloud Operations.

[17] Venkitaraman, A. K., & Kosuru, V. S. R. (2024). A hybrid technique for an autonomous vehicle control system to enhance the vehicle robustness: a HBA-RBFNN technique. International Journal of Electric and Hybrid Vehicles, 16(2), 103-135.

[18] Praveen, M. R., Kuchhal, P., & Choudhury, S. (2024). Novel statistical method for data drift detection in satellite telemetry. International Journal of Communication Systems, 37(9), e5766.

[19] Joshi, H. C., & Kumar, S. (2024). Artificial intelligence failures in autonomous vehicles: Causes, implications, and prevention. Computer, 57(11), 18-30.

[20] Hassan, N. A. (2024). Big Data and Machine Learning in Autonomous Vehicle Navigation: Challenges and Opportunities. Journal of Applied Cybersecurity Analytics, Intelligence, and Decision-Making Systems, 14(12), 54-64.

Received: Aug 11, 2025