Artificial Intelligence and Machine Learning for Intelligent DNS Threat Detection

Aakash Patel
Research Scholar
Vidhyadeep University
Kim, Surat
Email: aakashp1881@gmail.com
ORCID: 0009-0004-9987-5647

Hemangini Patel
Assistant professor
Vidhyadeep University
Kim, Surat
Email: hemangini.vimat@gmail.com
ORCID: 0009-0008-9040-2324

Sapna Patel
Assistant professor
Vidhyadeep University
Kim, Surat
Email: sapna.sp9404@gmail.com
ORCID: 0009-0008-0291-3627

Abstract- The Domain Name System (DNS) is a fundamental component of Internet communication and has increasingly become a target and communication channel for malicious activities. Machine learning (ML), deep learning (DL), and other artificial intelligence (Al) techniques have been investigated for detecting malicious domains, Domain Generation Algorithm (DGA) activity, botnet behavior, DNS exfiltration, and encrypted DNS traffic. However, the reviewed research addresses these problems using different data representations, datasets, learning models, and deployment assumptions. This review analyzes a selected corpus of 15 studies covering supervised and unsupervised learning, sequence-based deep learning, graph-based detection, explainable artificial intelligence (XAI), adversarial learning, encrypted DNS analysis, scalable DNS processing, and in-network threat mitigation. The literature is examined with respect to threat coverage, feature and traffic representation, learning methodology, datasets, evaluation, explainability, robustness, and deployment considerations. The analysis identifies several persistent challenges, including fragmented detection of different DNS threats, dependence on specific datasets, limited generalization to evolving attacks, reduced visibility in encrypted DNS, insufficient operational explainability, vulnerability to adversarial manipulation, and challenges associated with scalable real-time deployment. Based on these findings, a conceptual unified Al-driven DNS security framework is proposed to integrate specialized Al-based detection, risk assessment, explainability, and web-based security monitoring. The framework is intended as a research direction for combining Python-based Al processing with a MERN-based monitoring layer and requires future experimental validation.

Keywords- DNS security, artificial intelligence, machine learning, deep learning, malicious domains, domain generation algorithms, DNS tunneling, DNS exfiltration, DNS-over-HTTPS, explainable Al.

I. INTRODUCTION

The Domain Name System (DNS) is a fundamental service of Internet communication because it provides the naming mechanism through which domain names are resolved to network addresses. The large volume and distributed nature of DNS communication also make it a valuable source of information for identifying suspicious network behavior. Research in the supplied literature demonstrates that DNS traffic can be analyzed for malicious-domain activity, botnet behavior, Domain Generation Algorithms (DGAs), DNS exfiltration, encrypted DNS communication, and other security-related anomalies [R1]-[R5].

The security relevance of DNS is increased by the fact that attackers can exploit legitimate DNS functionality as part of their infrastructure. Malicious domains can be used for command-and-control communication, while DGA-based malware can generate large numbers of domains to make malicious infrastructure more difficult to identify using static indicators [R1], [R2]. DNS can also be abused as a channel for tunneling or exfiltrating

information, allowing data to be transferred through DNS requests and responses [R6], [R7]. These different forms of abuse produce different observable characteristics, which makes a single detection strategy difficult to apply across all DNS threats.

Artificial intelligence and machine learning have therefore become important approaches for DNS-security analysis. The reviewed studies include conventional machine-learning classifiers, deep sequence models, unsupervised learning, graph-based learning, and explainable Al techniques [R1]-[R1O]. These approaches operate on different representations of DNS activity. Domain-name strings can provide lexical information for DGA detection, statistical and behavioral characteristics can support exfiltration or bot net analysis, and graph representations can capture relationships among users, domains, hosts, or other DNS entities [R2], [R3], [R5].

Deep learning provides additional capability for learning complex representations from DNS and network data. Sequence-based models can capture patterns within domain names, while autoencoder-based approaches can support detection when complete malicious labels are unavailable [R1], [R2]. Graph-based approaches provide another perspective by modeling relationships among DNS entities and can therefore capture contextual information that may not be available from an isolated DNS query [R3], [R5]. These approaches demonstrate the increasing diversity of Al techniques being applied to DNS security.

However, the literature also reveals limitations that extend beyond the choice of machine-learning algorithm. The performance of a DNS detector can depend strongly on the dataset, feature representation, attack families represented during training, and evaluation environment. A model that performs effectively on a particular dataset may require further validation before its performance can be generalized to different DNS environments or previously unseen threats [R1], [R2], [R4]. The availability of labeled DNS data is itself an important research issue, as demonstrated by work addressing labeled DNS datasets for botnet detection [R4].

Explainability is another important consideration for security-oriented Al systems. High detection performance does not necessarily provide a security analyst with sufficient information to understand why a domain or DNS event was classified as malicious. The supplied literature therefore investigates explainable Al methods for DNS-related detection, including the use of techniques such as SHAP and other explanation mechanisms [R2], [R7]. Explainability can help connect model predictions with interpretable DNS characteristics and may support investigation and response.

The security of the Al model itself must also be considered. The reviewed adversarial-learning literature demonstrates that malicious-domain detectors, including graph-based models, can be exposed to attacks designed to manipulate model behavior [R9]. This introduces a second security problem: a DNS-security system must not only detect malicious activity but should also be resilient to attempts to deceive the

detection model.

Encrypted DNS introduces an additional challenge because protocols such as DNS-over-HTTPSreduce the visibility available to systems that depend on direct inspection of conventional DNS information. The supplied literature investigates flow-based and deep-learning approaches for identifying encrypted or encoded traffic and for analyzing DoH-related threats [R1O]-[R12]. These studies indicate that security monitoring must increasingly consider traffic-flow, temporal, and behavioral information when DNS content is not directly available.

Finally, practical deployment requires more than an accurate classification model. DNS generates substantial volumes of traffic, creating requirements for efficient data acquisition, processing, storage, inference, and monitoring. The supplied literature includes work on scalable DNS data processing and programmable-network approaches for in-network machine-learning-based mitigation [R13], [R14]. These studies demonstrate that data and deployment architecture are important components of an operational DNS-security solution.

Taken together, the reviewed literature represents significant progress in applying Al to individual DNS-security problems, but the approaches remain distributed across different threats, datasets, learning paradigms, and operational requirements. This fragmentation motivates a broader analysis of how malicious-domain detection, DGA detection, DNS tunneling and exfiltration, encrypted DNS analysis, explainability, adversarial robustness, and scalable deployment can be considered within a common security architecture.

The objective of this review is therefore to systematically synthesize the selected 15 studies and identify the technical and operational gaps that remain across Al-based DNS threat detection. Based on these findings, the review proposes a conceptual unified Al-driven DNS security framework that combines specialized Al detection components with risk assessment, explainability, and a MERN-based monitoring layer. The proposed framework is presented as a conceptual research direction and not as an experimentally validated system.

II. REVIEW METHODOLOGY

A. Review Scope

This review examines the application of artificial intelligence (Al), machine learning (ML), and deep learning (DL) to DNS security using a selected corpus of 15 research papers. The reviewed studies cover multiple DNS-security perspectives, including malicious-domain detection, Domain Generation Algorithm (DGA) detection, DNS botnet behavior, DNS tunneling and exfiltration, encrypted DNS and DNS-over-HTTPS (DoH), graph-based DNS analysis, explainable artificial intelligence (XAI), adversarial robustness, scalable DNS processing, and in-network mitigation.

The purpose of the review is to synthesize these studies according to their security objectives and technical characteristics rather than treating each paper as an independent solution. This allows the literature to be examined across common dimensions and helps identify where existing approaches complement one another and where important limitations remain.

B. Analytical Criteria

The selected studies are analyzed using a common set of criteria: **security threat, data or feature representation, AI/ML/DL technique, dataset, evaluation approach, explainability, robustness, scalability, and deployment considerations.** Threat and data representation are considered first to determine what type of DNS evidence is used by each approach. Model characteristics are then examined to understand

whether the study relies on conventional machine learning, deep learning, unsupervised learning, graph-based learning, or other specialized Al techniques.

Dataset and evaluation characteristics are considered separately because reported detection performance is closely related to the data used for training and testing. For example, the reviewed DNS-exfiltration research uses the CIC-Bell-DNS-EXF-2021 dataset and evaluates multiple classification methods, while other studies use domain-name collections, DNS logs, network-flow data, or specially generated encrypted-DNS traffic [1], [2]. The review therefore considers dataset scope and evaluation conditions when comparing reported results rather than treating performance values from different experimental settings as directly equivalent.

The analysis additionally considers explainability and robustness because a high-performing detector may still be difficult to operate or vulnerable to manipulation. XAl-oriented studies examine techniques for interpreting model predictions, while adversarial research demonstrates that Al-based malicious-domain detectors can themselves become targets of attack [3], [4]. Finally, scalability and deployment are considered because DNS produces high-volume traffic and practical security monitoring requires efficient data acquisition, processing, inference, and alert management [5],[6].

C. Literature Synthesis Approach

The literature is synthesized thematically across related DNS-security problems. Studies addressing similar threats are compared to identify common approaches, differences in data representation, and limitations in their experimental or operational settings. DGA-related studies, for example, are considered together to examine the progression from domain-feature analysis toward deep sequence learning, unsupervised detection, and explainability. Similarly, DNS-exfiltration studies are examined together to compare machine-learning classification, neural-network detection, and XAl-based analysis.

The same comparative approach is applied to graph-based DNS analysis, encrypted-DNS detection, adversarial robustness, and scalable deployment. This thematic synthesis provides the basis for the comparative analysis and research-gap identification presented in Sections 111-V. The objective is not to claim that one learning technique is universally superior, but to determine which capabilities are addressed by the existing literature and which capabilities remain fragmented across separate approaches.

TABLE I

SUMMARY OF THE SELECTED DNS SECURITY LITERATURE

Security Problem	Al/ML Approach	Data/Feature Focus	Main Research Focus
DGA detection	LSTM with attention	Domain-name sequences	Deep-learning-based DGA detection
DNS exfiltration	ML classification	DNS traffic features	Comparative exfiltration detection
DNS exfiltration	DNN with XAI	DNS-exfiltration features	Explainable threat detection

Malicious-domain detection	GNN / adversarial analysis	DNS graph features	Backdoor vulnerability of GNN-based detection
DNS traffic processing	Scalable data acquisition	DNS traffic and enriched metadata	High-volume DNS processing
DNS threat mitigation	ML with P4	DNS traffic	In-network ML offloading
DGA/botnet detection	Unsupervised deep learning	Linguistic and statistical domain features	Unsupervised detection and SHAP-based feature engineering
Encrypted DNS/ DoH	Flow-based ML	Standard network-flow telemetry	DoH detection
Encoded traffic	Hybrid CNN-LSTM	Network traffic	Encoded and malicious traffic characterization
DoH threat detection	Neuromorphic and adversarial learning	Encrypted DoH traffic	Robust and privacy-aware DoH detection
DNS behavioral risk	Graph-based learning	User-domain/DNS relationships	User and domain risk assessment
DNS configuration problems	Graph-based deep learning	DNS resolution traces/configuration graph	DNS misconfiguration diagnosis
DNS botnet detection	ML classification	Labeled DNS features	Reproducible DNS botnet detection
Malicious-domain detection	Graph-based learning	DNS/domain relationships	Structural malicious-domain analysis
DNS security analysis	ML/DL-based detection	DNS/network characteristics	Complementary DNS threat analysis

Ill. LITERATURE SURVEY: Al-BASED DNS THREAT DETECTION

A. Malicious Domain and DNS Behavioral Detection

Malicious-domain detection is one of the primary applications of DNS-based security analytics because domains can be used to support command-and-control communication, malware distribution, phishing, and other malicious activities. The reviewed literature shows two broad directions: approaches that classify domains using local characteristics and approaches that incorporate relationships among DNS entities.
Feature-based methods analyze properties such as domain characteristics, DNS behavior, temporal information, or resolution-related attributes, whereas graph-based methods attempt to capture relationships between domains, IP addresses, users, and other DNS entities [4],[11], [14].

The Tl-16 DNS study demonstrates the importance of labeled DNS data for botnet-oriented machine-learning research. Its analysis considers DNS features and evaluates machine-learning classification for DGA-related bot detection, while also highlighting the role of publicly available datasets in reproducible security research [13]. Such datasets provide a basis for model development and comparison, but the characteristics of a particular dataset can influence the behavior and reported performance of a detector.

Graph-based approaches extend conventional feature-based detection by incorporating relationships among DNS entities. The reviewed graph-based risk-assessment study uses DNS observations to derive user and domain risk information and demonstrates how DNS behavior can be considered in a broader behavioral context [11]. Similarly, malicious-domain detection through graph inference uses relationships among domains and related DNS entities to strengthen the identification of suspicious domains [14]. DNScope applies graph-based deep learning to DNS configuration analysis, demonstrating that graph representations can also be useful for identifying structural problems in DNS resolution systems [12].

The distinction between local and relational information is important for an intelligent DNS-security architecture. Local features can support efficient classification of individual DNS observations, while relational information can provide additional context when maliciousness depends on interactions among multiple entities. However, graph-based approaches also introduce additional requirements for graph construction, feature propagation, and continuous updating as DNS relationships change [11], [12], [14].

Therefore, the literature indicates that malicious-domain detection is moving beyond isolated domain classification toward behavioral and relational analysis. Nevertheless, these approaches remain distributed across different representations and objectives, leaving an opportunity for a unified architecture that can combine domain-level, behavioral, and graph-level evidence.

B. Domain Generation Algorithm Detection

Domain Generation Algorithms (DGAs) allow malware to generate large numbers of domain names, from which a smaller subset can be used to establish communication with command-and-control infrastructure. The constantly changing domain space created by DGA-based malware makes static blacklists and manually maintained indicators less suitable for detecting previously unseen domains. The reviewed literature consequently applies machine learning and deep learning to identify patterns associated with generated domain names [1], [7], (13].

The AdamW+ study investigates deep-learning-based DGA detection using domain-name text classification and compares adaptive optimization strategies, including Adam, AdamW, and the proposed AdamW+ optimizer [1]. The study illustrates how sequential or textual characteristics of domain names can be exploited by deep-learning models. Such approaches can learn complex patterns that are difficult to capture through manually designed rules.

A different direction is presented by the unsupervised DNS botnet detector based on deep autoencoders and explainable Al [7]. Instead of depending entirely on labeled DGA families, the approach trains autoencoders using unlabeled domain-name data and uses statistical and linguistic characteristics to distinguish legitimate names from anomalous DGA-generated names. SHAP is then used to interpret model behavior and guide feature augmentation. This approach addresses two practical difficulties associated with supervised DGA detection: the cost of obtaining reliable labels and the rapid emergence of new DGA families [7].

The Tl-16 dataset study provides complementary evidence from the data perspective. Its labeled DNS dataset supports botnet and DGA-oriented analysis and demonstrates the importance of DNS data containing sufficient contextual and labeling information for machine-learning research [13]. However, the availability of labeled examples does not remove the generalization problem. A detector trained on known DGA families may require additional validation when new families or substantially different DNS environments are encountered.

The reviewed DGA literature therefore shows a progression from feature-oriented detection toward deep sequence learning and unsupervised representation learning. Supervised deep models can learn highly discriminative patterns when representative training data are available, while unsupervised approaches can reduce dependence on exhaustive family-level labeling [1], [7]. The remaining challenge is to maintain reliable detection when DGA behavior changes and when the operational data differ from the training environment.

C. DNS Tunneling and Exfiltration Detection

DNS tunneling and DNS data exfiltration exploit DNS communication to transfer information through a protocol that is normally associated with name resolution. The reviewed literature shows that such attacks can be difficult to identify because malicious requests may resemble legitimate DNS activity and because large DNS volumes can make anomalous behavior difficult to isolate [2],[3].

The CIC-Bell-DNS-EXF-2021 study evaluates several machine-learning models for DNS data-exfiltration detection, including Random Forest, K-Nearest Neighbor, Gradient Boosted Trees, and Decision Tree approaches [2]. The study also considers both stateful and stateless characteristics and integrates model selection with a multi-criteria decision-making process for security-information and event-management applications. This demonstrates that DNS-exfiltration detection is not solely a model-selection problem; the operational context and characteristics of the detection system also influence how the models can be used.

The XAl-oriented DNS-exfiltration study extends this problem by examining how neural-network predictions can be interpreted by security analysts [3]. The work compares global and local explanation techniques, including SHAP, permutation feature importance, accumulated local effects, LIME, and Anchor. Its findings illustrate that different explanation methods involve different trade-offs between interpretability and computational requirements [3].

The combination of these studies highlights two complementary requirements. First, DNS-exfiltration detection requires features capable of distinguishing malicious DNS behavior from legitimate high-volume or unusual DNS activity. Second, the resulting predictions should be sufficiently interpretable to support security investigation. A model that produces a high-confidence alert without an understandable explanation can limit the practical value of the detection system.

The literature also indicates a distinction between stateless and stateful detection. Stateless approaches can provide efficient classification from individual observations or aggregated features, while stateful approaches can incorporate temporal and behavioral context [2]. The latter can provide additional information for sophisticated attacks but may require more resources and more complex processing. This trade-off becomes

particularly important when the goal is to integrate detection into a real-time security-monitoring architecture.

D. Encrypted DNS and DNS-over-HTTPS Detection

The deployment of encrypted DNS, particularly DNS-over-HTTPS (DoH), introduces a significant visibility challenge for DNS security. DoH encrypts DNS communication within HTTPS, which improves privacy but also prevents security systems from relying on direct inspection of conventional DNS content. The reviewed literature therefore investigates alternative representations, particularly network-flow characteristics and learned representations of encrypted traffic [8]-[1O].

The DoH detection study using standard flow telemetry demonstrates that encrypted DNS can be identified using IP-based and flow-level characteristics together with machine learning and active probing [8]. An important contribution of this approach is its emphasis on standard flow-monitoring infrastructure rather than specialized monitoring systems. This makes the approach relevant to practical deployment in networks that already use flow-monitoring equipment [8].

The hybrid deep-learning study addresses encoded network traffic using a combination of convolutional neural networks and long short-term memory networks [9]. The CNN component is used to extract useful information from traffic representations, while the LSTM component captures longer-term dependencies. The study further evaluates the approach using multiple cybersecurity datasets, demonstrating the potential of hybrid deep learning for classifying encoded traffic and distinguishing benign, zero-day, and malicious behavior [9].

The NQAL study explores a more advanced direction by combining neuromorphic, adversarial, privacy-oriented, and encrypted-traffic analysis concepts for DoH threat detection [1O]. Its discussion of previous encrypted-traffic approaches identifies limitations involving traffic padding, timing-based evasion, model interpretability, computational cost, false positives, privacy, and scalability [10]. These limitations reinforce the observation that encrypted-DNS security cannot be treated simply as a conventional classification problem.

An important distinction must therefore be maintained between **detecting DoH traffic** and **detecting malicious activity carried through DoH.** Flow-based classification can establish whether traffic exhibits characteristics associated with DoH, but additional behavioral or threat-analysis mechanisms may be required to determine whether the communication represents a security threat. A unified DNS-security architecture should consequently treat encrypted-DNS detection as one component of a broader threat-analysis pipeline rather than as a complete security solution.

E. Graph-Based and Structural DNS Intelligence

Graph-based learning provides a complementary representation of DNS activity by modeling relationships among domains, IP addresses, users, DNS configurations, and other entities. Unlike approaches that treat DNS events independently, graph-based methods can use information propagated through connected entities to improve contextual analysis [11], [12], [14].

The reviewed user-centric risk-assessment study demonstrates how DNS query relationships can be used to derive risk information for domains and users or network segments [11]. Its approach illustrates the value of combining domain-level threat information with user behavior rather than treating every DNS request as an isolated event. This provides a potential foundation for risk-oriented security monitoring.

Graph-based malicious-domain detection takes a similar relational perspective but focuses more directly on identifying suspicious domains through relationships in DNS data [14]. Such approaches can benefit from the

observation that malicious infrastructure may exhibit relationships that are not apparent from the characteristics of a single domain. However, the dependence on graph structure also creates an additional attack surface for the learningsystem itself.

The STING study provides important evidence of this limitation by demonstrating a backdoor attack against GNN-based malicious-domain detection. The study shows that an attacker can manipulate selected graph features and edges to implant triggers that influence model predictions while maintaining relatively small changes to legitimate-domain performance [4]. This finding is significant because it changes the security question from simply asking whether graph learning improves detection to also asking whether the graph-based detector can itself be trusted under adversarial conditions.

DNScope extends graph-based reasoning to DNS configuration analysis rather than malicious-domain detection [12]. Its use of graph-based deep learning demonstrates that structural DNS information can be applied to problems beyond conventional threat classification. Taken together, these studies suggest that graph representations can provide valuable contextual intelligence, but their use must be accompanied by consideration of graph quality, dynamic updates, computational cost, andadversarial robustness.

Overall, the literature demonstrates that no single representation fully captures the DNS-security problem. Domain-level features are useful for DGA and malicious-domain analysis, traffic features are important for tunneling, exfiltration, and encrypted DNS, while graph representations provide relational context. This observation provides a technical basis for the unified framework proposed later in this review.

TABLE II

COMPARATIVE ANALYSIS OFAI-BASED DNS THREAT DETECTION

Threat	Features/Data	Model/Approach	Dataset/Data Source	Reported Focus	Key Limitation
DGA	Domain-name/text features	Deep learning with Adam, AdamW, and AdamW+	DGA domain data	Optimization for DGA detection	Focused primarily on model/optimizer improvement
DGA/Botnet	Linguistic and statistical domain features	Deep Autoencoder + SHAP	Tranco/DGArchive-based data	Unsupervised detection and feature augmentation	Evolving/adversarial DGA remains challenging
DGA/Botnet	Labeled DNS features	ML classification	TI-16 DNS dataset	Labeled DNS botnet research	Dataset dependence
DNS behavioral risk	User-domain relationships	Graph-based risk learning	CIRA DNS Firewall logs	User/domain risk assessment	User identification can be affected by NAT
Malicious Domains	DNS graph relationships	Graph-based learning	DNS/domain graph data	Relational malicious-domain detection	Graph dependency and robustness
DNS exfiltration	DNS traffic features	RF, KNN, GBT, DT + MCDM	CIC-Bell-DNS-EXF-2021	Comparative exfiltration detection	Dataset-specific evaluation
DNS exfiltration	DNS-exfiltration features	DNN + XAI	CIC-Bell-DNS-EXF-2021	Explainable detection	XAI computational trade-offs
DoH	IP/flow features	ML + active probing	Lab and real-world ISP data	DoH classification	DoH detection does not by itself establish maliciousness

Encoded traffic	Network traffic	CNN + LSTM	SCADA/NSL-KDD/CICIDS2017	DoH/Non-DoH and threat characterization	Evaluation context differs from general Internet DNS
DoH threats	Encrypted traffic	Advanced adversarial/neuromorphic learning	Encrypted DoH data	Robust encrypted-DNS analysis	High architectural complexity
Malicious DNS domains	DNS graph features/edges	GNN + adversarial analysis	Real-world GNN-MDD settings	Backdoor vulnerability analysis	Demonstrates detector vulnerability
DNS configuration	Configuration/resolution graph	Graph neural learning	DNS configuration data	Misconfiguration diagnosis	Different objective from malicious-domain detection

IV. EXPLAINABLE, ROBUST, AND DEPLOYABLE DNS SECURITY

The effectiveness of an Al-based DNS security system cannot be evaluated only by its classification performance. A security-oriented model must also provide interpretable decisions, remain reliable when its inputs are manipulated, operate across suitable datasets, and process DNS traffic within practical resource and latency constraints. The reviewed literature addresses these requirements through explainable artificial intelligence, adversarial analysis, dataset-oriented evaluation, scalable data processing, and deployment-oriented architectures.

A. Explainable Al for DNS Security

Explainability is particularly important in DNS security because model predictions can directly influence the investigation of domains, hosts, and network activity. A prediction that identifies a DNS event as malicious is more useful to an analyst when the system can also indicate which characteristics contributed to that decision. The reviewed literature contains several approaches for incorporating explainability into DNS-related machine-learning and deep-learning systems [3], [7].

The unsupervised DNS botnet detection study integrates SHAP with deep autoencoder-based detection and uses feature importance to guide feature augmentation [7]. This demonstrates that explainability can contribute not only to post-hoc interpretation but also to understanding which statistical and linguistic characteristics are influential in the detection process. Such information can be valuable when assessing whether a model is learning meaningful DNS characteristics or relying on potentially unstable features

The DNS-exfiltration XAI study provides a broader comparison by evaluating six explanation techniques: global SHAP, permutation feature importance (PFI), accumulated local effects (ALE), local SHAP, LIME, and Anchor [3]. The study demonstrates that explanation methods differ in their computational requirements and interpretability characteristics. Global methods can help identify features that influence the model across a dataset, whereas local explanations can help explain an individual prediction.

For DNS security operations, this distinction is important. A security analyst may need both a general understanding of model behavior and an explanation for a particular alert. An integrated system should therefore treat XAI as part of the detection workflow rather than as an independent visualization feature. In the proposed framework, XAI is consequently positioned after threat prediction so that an alert can be accompanied by information describing the factors that contributed to the model's decision.

However, explainability does not automatically establish model correctness. An explanation describes the relationship between the model and its prediction, but the explanation itself must still be interpreted within the characteristics and limitations of the underlying dataset and model. Therefore, XAI should complement, rather

than replace, model evaluation and security validation.

B. Adversarial Robustness

Al-based DNS security introduces a security dependency on the detector itself. Attackers may attempt to modify DNS inputs or manipulate the data representation used by a model so that malicious activity is classified as benign. The reviewed literature provides evidence that this issue is particularly relevant to graph-based malicious-domain detection [4].

The STING study investigates a stealthy backdoor attack against graph neural network-based malicious-domain detection through DNS perturbations [4]. The study demonstrates that manipulating graph-related information can influence the behavior of a GNN detector while attempting to preserve the normal performance of the model on non-triggered data. This illustrates that graph-based DNS security systems require security analysis beyond conventional accuracy evaluation.

Adversarial robustness is also relevant to encrypted-DNS detection. The NQAL study incorporates adversarial learning into a broader architecture for DNS-over-HTTPSthreat detection and considers the difficulty of detectingtraffic when attackers can alter observable characteristics such as timing and traffic patterns [1O].
This direction indicates that encrypted-DNS security requires consideration of both limited visibility and intentional evasion.

These findings suggest that robustness should be treated as a separate evaluation dimension in Al-driven DNS security. A detector should not be considered sufficiently secure solely because it achieves high performance on a static test set. Future evaluations should examine how predictions change when relevant DNS features, graph relationships, or observable traffic characteristics are deliberately modified.

For the proposed framework, adversarial robustness is therefore considered a design requirement rather than a claim of an already validated capability. Robustness mechanisms and adversarial testing would need to be implemented and evaluated experimentally before the proposed system could be considered resistant to such attacks.

C. Dataset and Evaluation Challenges

Dataset selection is a central issue in DNS-security research because different studies operate on substantially different forms of data. The reviewed literature includes labeled DNS datasets, domain-name collections, DNS firewall logs, DNS-exfiltration traffic, network-flow telemetry, encrypted traffic, and DNS configuration information [2], [7], [8], [11], [12], [13]. These differences make direct numerical comparison between studies difficult.The Tl-16 study demonstrates the value of labeled DNS data for reproducible botnet and DGA research [13]. In contrast, the unsupervised DNS botnet study addresses situations in which extensive labeling is unavailable by learning representations from largely unlabeled domain data [7]. These approaches represent different points on the supervision spectrum and demonstrate that dataset availability can influence the choice of learning methodology.

DNS-exfiltration research provides another example of dataset-specific evaluation. The CIC-Bell-DNS-EXF-2021-based studies evaluate models using a particular set of DNS-exfiltration characteristics [2], [3]. Although such evaluation is useful for comparing models under a common experimental setting, performance obtained on one dataset should not automatically be interpreted as evidence that the same model will perform identically on unrelated DNS environments.

The same issue is visible in encrypted-DNS research. Flow-based DoH detection uses network telemetry rather than direct DNS content [8], while hybrid deep-learning approaches use representations of encoded

traffic [9]. Consequently, differences in traffic environment, feature construction, and data collection can affect the resulting evaluation.

Evaluation metrics must also be interpreted carefully. Accuracy can be informative when classes are balanced, but DNS-security datasets may contain substantially different proportions of benign and malicious observations. Precision, recall, F1-score, false-positive rate, and detection latency can therefore provide additional information about operational performance. Cross-dataset evaluation would provide stronger evidence of generalization than evaluation on a single dataset alone.

The literature consequently indicates that dataset diversity and evaluation design are as important as model selection. A future unified DNS-security system should evaluate individual detection components using their relevant datasets while also investigating whether the learned representations remain useful under different network conditions.

D. Scalability, Privacy, and Deployment

A DNS detector that performs well in an offline experiment may still face significant challenges when deployed on high-volume traffic. DNS generates a continuous stream of queries and responses, requiring efficient acquisition, preprocessing, feature extraction, inference, storage, and alert delivery. The reviewed literature contains dedicated work addressing these infrastructure requirements [5], [6].

The scalable DNS data-acquisition study presents an architecture designed to process DNS traffic efficiently while supporting enrichment, anonymization, and high-volume storage and analysis [5]. This work demonstrates that the data pipeline surrounding an Al model can become a significant part of the overall system architecture. Efficient collection and preprocessing are therefore necessary before an ML model can be applied reliably at scale.

The P4DME work explores another deployment direction by moving machine-learning-based DNS threat mitigation toward programmable network infrastructure [6]. Its focus on in-network processing addresses the possibility that sending all traffic to centralized detection servers can introduce processing and throughput bottlenecks. Although such approaches are more specialized than a conventional web-based monitoring system, they demonstrate the broader requirement for low-latency and scalable DNS-security processing.

Privacy is another consideration because DNS data can contain information about users, devices, and requested services. The reviewed graph-based and scalable DNS studies include privacy-related processing and anonymization considerations [5], [11]. A practical monitoring architecture should therefore limit unnecessary exposure of identifying information and define appropriate data-retention and access policies.

These deployment studies suggest that an Al-driven DNS-security system should be designed as more than a standalone prediction model. Data acquisition, processing, model inference, alert management, storage, visualization, and privacy controls must operate as connected components. The proposed framework adopts this system-level perspective by separating Python-based Al inference from the MERN-based monitoring and application layer.

TABLE III

DATASET, EXPLAINABILITY, ROBUSTNESS, AND DEPLOYMENT ANALYSIS

Ref.	Dataset / Data Source	Threat / Task	XAI	Adversarial Evaluation / Analysis	Focus	Deployment Consideration
1.	CIC-Bell-DNS-EXF-2021	DNS exfiltration	No primary XAI focus	Not primary focus	ML/MCDM comparison	SIEM-oriented detection
2.	CIC-Bell-DNS-EXF2021	DNS exfiltration	SHAP, PFI, ALE, LIME, Anchor	Not primary focus	Explainability comparison	Computational cost of XAI
3.	DNS graph data	Malicious-domain detection	Not primary focus	GNN backdoor attack and model behavior	Robustness requirement	Robustness requirement
4.	Large-scale DNS traffic	DNS data processing	Not primary focus	Not primary focus	Processing efficiency	Scalable acquisition and storage
5.	DNS traffic	Threat mitigation	Not primary focus	Not primary focus	In-network ML feasibility	P4-based processing
6.	Domain/DGA datasets	DGA/botnet detection	SHAP	Adversarial limitations identified	Unsupervised detection	Scalability/privacy considerations
7.	Lab and real-world flow data	DoH detection	Not primary focus	Not primary focus	Flow-based classification	Standard telemetry
8.	SCADA/NSL-KDD/CICIDS2017-related traffic	Encoded/malicious traffic	Not primary focus	Not primary focus	Hybrid DL characterization	Computational complexity
9.	Encrypted DoH traffic	DoH threat detection	Discussed as future requirement	Adversarial encrypted-traffic learning	Robust encrypted-traffic detection	Real-time/privacy considerations
10.	CIRA DNS Firewall data	User/domain risk	Not primary focus	Not primary focus	Risk assessment	Privacy/anonymization
11.	DNS configuration/resolution data	DNS misconfiguration	Not primary focus	Not primary focus	Graph-based diagnosis	Automated diagnosis
12.	TI-16 DNS	Botnet/DGA	Not primary focus	Not primary focus	Labeled ML	Reproducibility

	dataset	detection			evaluation	
13.	DNS/domain graph data	Malicious-domain detection	Not primary focus	Related graph-security concerns	Graph inference	Dynamic graph requirement
14.	DNS/network security data	DNS threat analysis	Varies by approach	Varies by approach	Threat detection	Application integration

V. COMPARATIVE ANALYSIS AND RESEARCH GAPS

The literature reviewed in the preceding sections demonstrates that artificial intelligence can support DNS security through multiple complementary forms of analysis. However, the studies generally optimize individual detection tasks, data representations, or deployment requirements rather than addressing the complete DNS-security lifecycle within a common architecture. This section therefore compares the reviewed approaches at the capability level and identifies the gaps that remain when these studies are considered collectively.

A. Cross-Study Comparative Analysis

The reviewed studies differ substantially in the type of evidence used for detection. Domain-oriented approaches primarily exploit lexical, linguistic, and statistical characteristics and are particularly relevant to DGA and malicious-domain detection [1], [7], [13]. Traffic-oriented approaches instead analyze DNS or network-flow characteristics and are more suitable for identifying exfiltration, tunneling, or encrypted DNS behavior [2], [8], [9]. Graph-based approaches provide an additional representation by modeling relationships between DNS entities, users, domains, or configuration elements [4], [11], [[12], [14].

These representations are complementary rather than interchangeable. A domain sequence can provide evidence that a name resembles a DGA-generated domain, but it does not necessarily reveal whether the corresponding host is participating in DNS exfiltration. Similarly, flow-based DoH detection can identify encrypted DNS behavior without necessarily establishing whether the communication is malicious [8]. Graph-based analysis can provide contextual relationships that are unavailable from an isolated DNS observation, but it requires construction and maintenance of an appropriate graph representation [11], [12], [14].

The reviewed models also vary in their learning paradigms. Conventional machine-learning methods provide comparatively direct classification pipelines and have been applied to DNS-exfiltration and other traffic-analysis problems [2]. Deep-learning methods provide greater representation-learningcapability for domain sequences, anomaly detection, and encoded traffic [1], [7], [9]. Graph-based learning introduces relational reasoning, while advanced adversarial approaches investigate the resilience of detection models against deliberate manipulation [4],[1O].

Another important difference concerns the level of operational maturity addressed by each study. Some research concentrates primarily on model performance, whereas other work explicitly considers scalable DNS data acquisition, network-level processing, privacy, or real-time deployment [5],[6],[1O], [11]. Consequently, a high detection score in one study should not automatically be interpreted as evidence that the approach is more suitable for operational deployment than a study with a lower reported score under a different experimental setting.

Explainability also remains uneven across the literature. The DNS-exfiltration XAI study explicitly compares multiple post-hoc explanation techniques and demonstrates that explanation quality and computational requirements vary among methods [3]. The unsupervised DGA study integrates SHAP into the feature-engineering process [7]. In contrast, explainability is not a primary objective of several other detection approaches. This indicates that interpretability is often treated as an additional capability rather than as an integral component of the detection pipeline.

Adversarial robustness presents a similar pattern. The STING study demonstrates a specific backdoor threat against GNN-based malicious-domain detection [4], while the NQAL study incorporates adversarial protection into an encrypted-DNS detection architecture [1O]. These studies show that robustness can be incorporated into Al-based DNS security, but they also indicate that robustness evaluation is not yet a uniform requirement across the broader literature.

B. What Existing Research Achieves

The reviewed literature provides several significant capabilities for DNS security. First, machine-learning approaches demonstrate that DNS and network-flow characteristics can be transformed into features suitable for automated threat classification [2], [8], [13]. Second, deep-learning approaches extend this capability by learning complex representations from domain names, unlabeled DNS data, and encoded network traffic [1], [7], [9].

Third, graph-based approaches demonstrate that DNS security can benefit from contextual relationships rather than relying exclusively on individual observations [11], [12], [14]. Fourth, XAI research provides mechanisms for interpreting model predictions and connecting automated decisions with characteristics that can be examined by security analysts [3], [7]. Fifth, encrypted-DNS studies demonstrate that traffic-flow and temporal information can provide useful security signals even when DNS content itself is encrypted [8]-[1O].

The literature also provides evidence that deployment considerations are receiving increasing attention. Scalable DNS data acquisition addresses the processing requirements associated with large DNS volumes [5], while P4DME investigates distributing detection between programmable network devices and ML servers to reduce processing bottlenecks [6]. These studies indicate that DNS security should be viewed as a system-level problem rather than solely as a classification problem.

C. What Remains Unresolved

Despite these advances, the reviewed studies reveal a fragmented capability landscape. A DGA detector may focus primarily on domain-name characteristics, an exfiltration detector may rely on DNS traffic features, a DoH detector may operate on encrypted network flows, and a graph-based detector may depend on relational DNS information. These specialized approaches are valuable, but their outputs are generally not combined into a common threat-assessment process.

A second issue is generalization. The reviewed studies use different datasets and experimental environments, including labeled DNS data, domain-name collections, DNS firewall logs, DNS-exfiltration datasets, network-flow data, encrypted traffic, and DNS configuration information [2], [5], [7], [8], [11], [12], [13]. Because these datasets represent different traffic populations and feature spaces, performance reported under one experimental condition cannot automatically be generalized to another.

A third challenge is the evolution of malicious behavior. The unsupervised DGA literature specifically identifies the difficulty of continuously labeling new DGA families and the need for models that can adapt to

changing threats[7].This problem is not limitedto DGA. Encrypted DNS traffic andadversarialmanipulation can also change the observable characteristicson whichdetection models depend[4],[1O].

A fourth gap concerns encrypted DNS. Flow-based DoH detection demonstrates that encrypted DNS can be identified using standard telemetry [8], but identifying DoH does not by itself determine whether the traffic contains malicious activity. Advanced encrypted-DNS research further shows that timing, padding, adversarial manipulation, privacy, and computational requirements can complicate detection [1O]. Thus, encrypted DNS requires a combination of visibility, behavioral analysis, and robustness rather than a single binary classifier.

A fifth gap concerns operational explainability. XAI methods can explain predictions, but the computational cost and level of detail vary among explanation techniques [3].Therefore, simply adding an XAI library to an existing model does not automatically create an operationally useful security system. Explanations need to be connected to alerts, threat categories, and analyst workflows.

Finally, scalability and deployment remain important challenges. High-volume DNS environments require efficient acquisition, processing, inference, storage, and response mechanisms [5], [6]. Centralized processing may create bottlenecks, while distributed or in-network processing introduces additional architectural and implementationrequirements.

D. Key Research Gaps

Based on the cross-study comparison, seven major research gaps are identified.

1) Fragmented multi-threat detection: Existing approaches generally focus on specific DNS threats or data representationsrather than providing a unified detection architecture covering multiple threat categories.

2) Limited cross-dataset generalization: Many studies evaluate their approaches within particular datasets or traffic environments, making broader generalization an important unresolved issue.

3) Evolving and previously unseen DGA behavior: Supervised DGA detection can require continuously updated labeled data, while unsupervised approaches still require validation against changing legitimate and malicious DNS behavior [7].

4) Reduced visibility in encrypted DNS: DoH limits direct inspection of DNS content, requiring security systems to rely on flow,temporal, and behavioral characteristics[8]-[1O].

5) Insufficient operational explainability: XAI can improve modeltransparency, but explanation methods have different computational and interpretability characteristics and are not consistently integrated into complete security workflows [3],[7].

6) Adversarial vulnerability: Al-based DNS detectors can themselves become targets of manipulation, as demonstrated by attacks against graph-based malicious-domain detection [4] and adversarial analysis of encrypted-DNS detection [1O].

7) Real-time scalability and deployment: High-volume DNS monitoringrequires efficient data acquisition, inference, storage, and response, while centralized processing can introduce resource and throughput constraints [5],[6].

These gaps indicate that the next research direction should not simply replace one classification algorithm with another. Instead, it should investigate how specialized detection models, heterogeneous DNS evidence, explainability, risk assessment, robustness, and deployment components can operate together within a unified

architecture.

TABLE IV.

OVERALL RESEARCH GAP ANALYSIS

Capability	Existing Research	Observed Limitation	Required Research Direction
Malicious-domain detection	ML, DL, and graph-based approaches	Different feature and graph representations	Combine complementary domain and behavioral evidence
DGA detection	Supervised DL and unsupervised learning	Evolving DGA families and labeling requirements	Adaptive and continual detection
DNS exfiltration	ML, DNN, and XAI approaches	Dataset and behavioral dependence	Context-aware detection with integrated explanation
Encrypted DNS / DoH	Flow-based and advanced DL approaches	Limited content visibility and evasion	Multi-signal encrypted-flow analysis
Graph-based detection	Graph learning and risk analysis	Graph construction and adversarial concerns	Robust dynamic graph analysis
Explainability	SHAP, LIME, PFI, ALE, Anchor	Computational and operational trade-offs	Integrated XAI for security alerts
Adversarial security	Backdoor and adversarial-learning studies	Robustness is not uniformly evaluated	Adversarially robust detection
Scalability	Scalable acquisition and in-network ML	Processing and throughput constraints	Distributed and scalable inference
Unified monitoring	Individual research solutions	Threat-specific and fragmented architectures	Unified AI-driven DNS security framework

VI. PROPOSED UNIFIED AI-DRIVEN DNS SECURITY FRAMEWORK

The comparative analysis indicates that existing AI-based DNS-security studies provide useful solutions for individual threat categories, but their detection capabilities, data representations, explainability mechanisms, robustness considerations, and deployment architectures are generally addressed separately. Based on these observations, this review proposes a conceptual unified AI-driven DNS security framework that combines complementary detection components into a common monitoring architecture. The framework is not presented as an experimentally validated system; rather, it represents a research direction derived from the limitations and capabilities identified in the reviewed literature.

A. Framework Objectives

The proposed framework has six primary objectives. First, it aims to support detection of multiple DNS threat categories rather than relying on a single threat-specific detector. Second, it separates threat detection

according to the type of evidence available, allowing domain-level, traffic-level, and behavioral information to be processed by appropriate models. Third, it incorporates risk fusion to combine outputs from different detection components into a common security assessment. Fourth, it introduces an explainability layer so that model predictions can be associated with interpretable evidence. Fifth, it considers robustness and future adversarial evaluation as part of the security architecture. Finally, it provides a web-based monitoring layer through which detected events and their associated risk information can be presented to security analysts.

B. Proposed Architecture

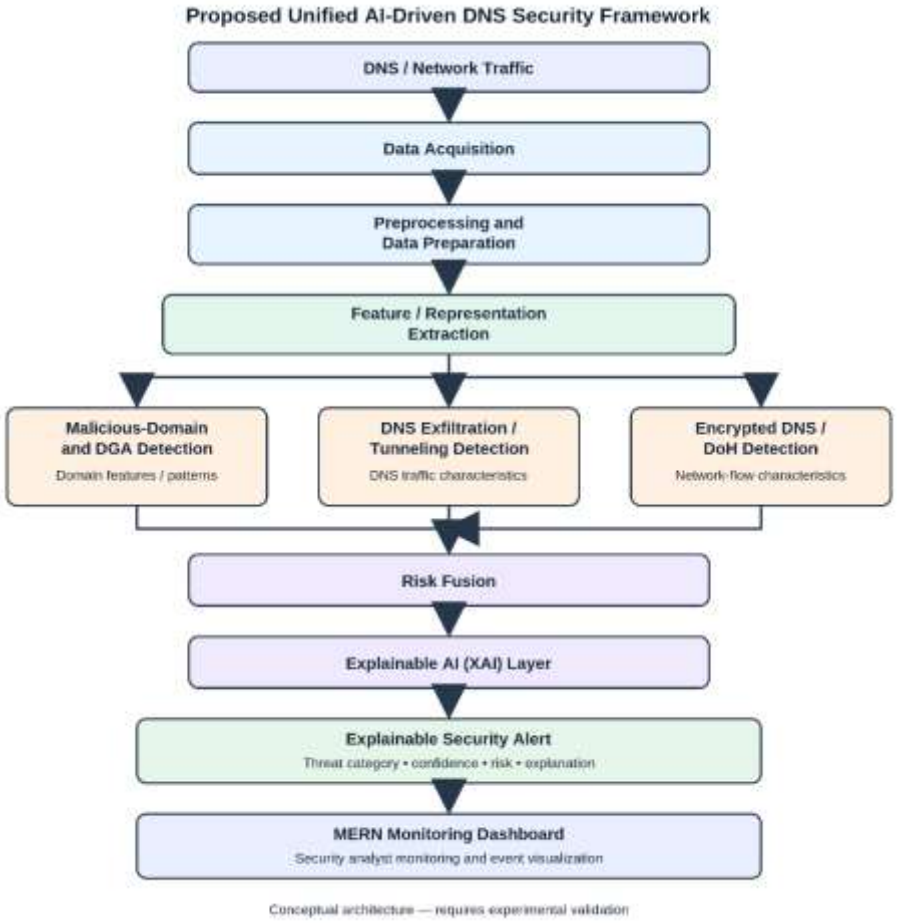
The proposed architecture consists of a data-acquisition layer, preprocessing layer, feature and representation extraction layer, specialized Al detection modules, risk-fusion layer, explainability layer, and monitoring layer.
DNS and related network observations first enter the data-acquisition component, where relevant traffic records are collected. The preprocessing stage then performs data cleaning, normalization, and preparation for the individual detection models.

After preprocessing, the framework separates the available evidence according to the detection task. Domain-oriented information can be provided to malicious-domain and DGA detection, DNS traffic characteristics can be used for tunneling and exfiltration analysis, and network-flow characteristics can be used for encrypted-DNS or DoH analysis. Graph-based information can additionally provide relational context where the required DNS relationships are available.

The outputs of the specialized detection components are subsequently passed to a risk-fusion layer. Rather than treating the output of a single model as the final security decision, the risk-fusion layer is intended to combine the available threat probabilities and contextual information. The resulting assessment is then passed to the XAI layer, where the factors contributing to the individual model predictions can be presented with the alert.

The final output is delivered to a monitoring layer that provides security analysts with information about the detected event, threat category, model confidence, risk assessment, and available explanation.

Fig. 1. Proposed unified Al-driven DNS security framework.



C. Al Detection Layer

The Al detection layer is designed as a collection of specialized models rather than a single universal classifier. This design follows the observation from the reviewed literature that different DNS threats expose different types of information. DGA detection can make use of domain-name characteristics and sequential patterns, DNS-exfiltrationdetection can usetraffic and statistical features, and encrypted-DNS detection can use network-flow characteristics [1], [2], [7], [8].

For the proposed implementation direction, Python can be used as the primary environment for data processing, modeltraining, and inference. Classical machine-learning models can be used where structured DNS or network-flow features are appropriate, while deep-learning models can be used for sequential or representation-learning tasks. The initial prototype can therefore use a threat-specific model strategy rather than attempting to force heterogeneous DNS observations into one model.

A possible implementation configuration is to use an LSTM-based model for DGA-oriented domain analysis, a classical machine-learningmodel such as Random Forest for DNS-exfiltration classification, and a flow-based machine-learningmodel for DoH detection. These choices are implementation directions rather than experimentally validated recommendations. Their suitability must be established through subsequent experiments usingthe selected datasets.

Graph-based analysis can be incorporated as an additional contextual component where suitable graph data are available. The reviewed literature demonstrates that graph representations can provide user-domain, domain-relationship,and DNS-configuration information that is not directly available from individual DNS observations

[4],[11], [12], [14].

D. Risk Assessment and Fusion Layer

The risk-fusion layer provides a common interface between the specialized detection models and the security-monitoring application. Each detector can produce a threat probability or confidence value together with its predicted threat category. These outputs can then be combined with contextual information to produce a unified security assessment.

For example, an event may produce a high DGA probability while showing limited evidence of DNS exfiltration. Another event may exhibit moderate malicious-domain probability together with strong evidence of suspicious DNS-flow behavior. The risk-fusion layer is intended to preserve these individual signals while providing a consolidated representation of the security state.

The exact mathematical formulation of the risk score is intentionally left for experimental design. A fixed weighting formula should not be introduced without evaluating the relative calibration and reliability of the individual models. Future implementation can investigate weighted fusion, calibrated probabilities, or learning-based fusion after the individual detectors have been evaluated.

E. Explainability Layer

The XAI layer is positioned after threat prediction and risk assessment so that the system can associate an alert with interpretable model evidence. SHAP can be used as the initial explanation mechanism because the reviewed literature demonstrates its application to DNS-related detection and feature analysis [3],[7].

For a domain-oriented alert, the explanation may identify influential domain or statistical characteristics. For a DNS-exfiltration alert, it may identify the traffic features that contributed most strongly to the prediction. For flow-based encrypted-DNS detection, the explanation can identify influential flow characteristics when the underlying model supports such interpretation.

The purpose of this layer is not to claim that an explanation proves that an event is malicious. Instead, it provides additional information about why the model generated a particular prediction. This distinction is important because XAI explains model behavior and does not replace independent security validation.

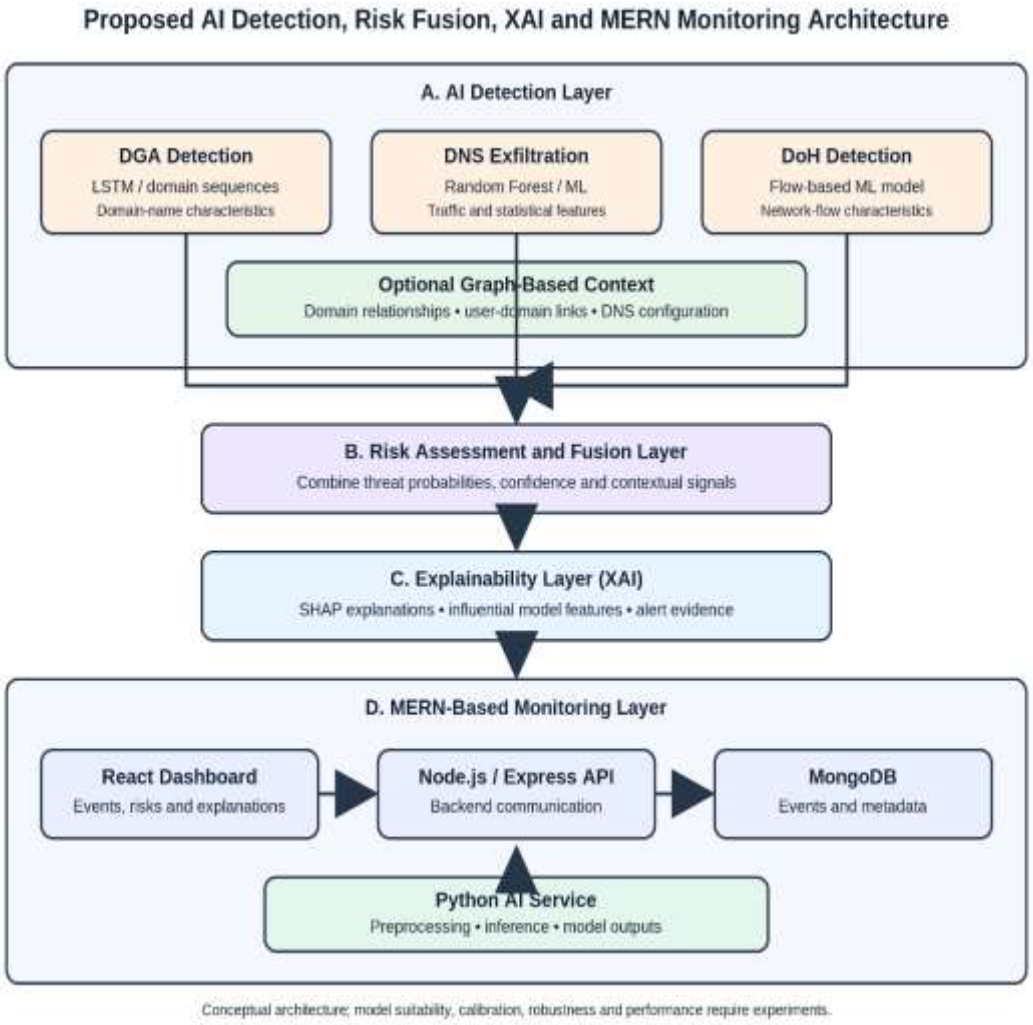
F. MERN-Based Monitoring Layer

The application and monitoring layer is designed around the MERN stack. React can provide the frontend dashboard through which security events, threat categories, confidence values, risk information, and explanations are displayed. Node.js and Express can provide the application programming interface (API) layer responsible for communication between the dashboard, database, and Al inference services.

MongoDB can be used to store DNS-security events and associated metadata, including timestamps, predicted threat categories, confidence values, risk assessments, and explanation information. Python can operate as a separate Al service responsible for preprocessing, model inference, and communication of prediction results to the application backend.

The resulting separation can be represented as:

React Dashboard



This architecture separates application development from AI inference. Consequently, a model can potentially be retrained or replaced without redesigning the complete monitoring interface. However, the scalability and latency of this architecture must be experimentally evaluated before claims about real-time performance are made.

G. Expected Contribution and Scope

The principal contribution of the proposed framework is architectural integration rather than the invention of a new AI algorithm. The framework brings together specialized DNS threat detection, risk fusion, explainability, and security monitoring within a common conceptual pipeline. It therefore addresses the fragmentation identified in the literature while preserving the specialized representations required by different DNS threats.

The framework also provides a practical direction for future implementation using Python-based AI processing and a MERN-based application layer. Nevertheless, several components remain research questions, including cross-dataset generalization, calibration of risk fusion, adversarial robustness, encrypted-DNS threat

characterization, inference latency, and large-scale deployment. These aspects require experimental investigation before the framework can be evaluated as an operational DNS-security system.

VII. FUTURE RESEARCH DIRECTIONS

The reviewed literature indicates that Al-based DNS security is developing rapidly, but several research directions remain important for improving adaptability, robustness, privacy, and practical deployment. Future work should therefore extend beyond improving the accuracy of individual detection models and investigate how Al-based DNS security can operate under changing threats and large-scale network conditions.

A. Continual and Adaptive Learning

DGA families and malicious infrastructure can change over time, creating a need for detection models that can adapt without requiring complete retraining whenever new attack patterns appear. The reviewed unsupervised DGA research demonstrates the difficulty of maintaining labeled datasets for continuously emerging DGA families [7]. Future research should therefore investigate continual and adaptive learning approaches that can incorporate new DNS observations while reducing degradation of previously learned detection capabilities.

B. Transformer and Graph-Based Hybrid Models

Future DNS-security systems can investigate combinations of sequential and relational representations. Transformer-based models may provide an alternative for learning long-range dependencies in domain names and DNS sequences, while graph-based models can capture relationships among domains, users, hosts, and other DNS entities [11], [14]. A hybrid architecture combining these representations could potentially provide broader contextual information than either representation alone. However, such approaches would require careful evaluation of computational cost, dataset requirements, and generalization.

C. Adversarially Robust DNS Detection

The existence of attacks against Al-based DNS detectors demonstrates the need for robustness evaluation as part of future DNS-security research [4]. Future studies should investigate adversarial training, robust feature representations, graph-structure protection, and systematic evaluation against evasion and backdoor attacks. For encrypted DNS, robustness should additionally consider manipulation of observable flow and temporal characteristics[1O].

D. Privacy-Preserving and Federated Learning

DNS traffic can contain information related to users, devices, and requested services. Privacy-preserving learning is therefore an important direction for environments in which DNS data cannot be freely centralized. Federated or distributed learning could allow multiple organizations or network segments to contribute to model development while limiting the direct exchange of raw DNS records. Such approaches would require further investigation of communication overhead, non-identically distributed DNS data, model privacy, and detection performance.

E. Real-Time Edge and In-Network Detection

The high volume of DNS traffic creates a need for efficient processing close to the point where traffic is generated or observed. The reviewed scalable data-processing and P4-based research demonstrates two

complementary directions for improving DNS-security processing efficiency [5], [6]. Future research can investigate lightweight Al inference at network edges, programmable-network acceleration, and distributed inference architectures. Evaluation should include not only detection performance but also inference latency, throughput, memory consumption, and operational overhead.

F. Explainable Al for Security Operations

Future work should move beyond generating model explanations and investigate how those explanations can support actual security-operations workflows. XAI techniques such as SHAP and other local or global explanation methods can provide information about model decisions, but their usefulness depends on whether the explanations are understandable, timely, and actionable for analysts [3], [7]. Integrating explanations with risk prioritization, alert correlation, and incident investigation represents an important direction for operationally usefulAl-based DNSsecurity.

VIII. CONCLUSION

Artificial intelligence, machine learning, and deep learning have demonstrated significant potential for improving DNS security across multiple threat categories. The 15 studies reviewed in this paper show that different learning paradigms can address different forms of DNS abuse, including malicious domains, Domain Generation Algorithms, DNS botnet behavior, DNS exfiltration, encrypted DNS, and DNS-related structural anomalies. Conventional machine learning provides effective approaches for structured DNS and traffic features, while deep learning, unsupervised learning, and graph-based methods provide additional capabilities for sequential, anomalous, and relational analysis.

The review also shows that detection performance alone is insufficient for evaluating an Al-driven DNS-security system. Dataset dependence, evolving attack behavior, limited visibility in encrypted DNS, explainability requirements, adversarial vulnerabilities, privacy considerations, and large-scale deployment requirements remain important challenges. The reviewed studies address many of these issues individually, but their capabilities remain distributed across different threat categories, data representations, and system architectures.

Based on the comparative analysis, seven major research gaps were identified: fragmented multi-threat detection, limited cross-dataset generalization, evolving and previously unseen DGA behavior, reduced visibility in encrypted DNS, insufficient operational explainability, adversarial vulnerability, and real-time scalability. These gaps indicate that future DNS-security research should move toward architectures that combine complementary detection methods rather than relying on a single model or threat-specific solution.

To address this direction, a conceptual unified Al-driven DNS security framework was proposed. The framework combines specialized Al-based detection for different DNS threat characteristics with risk fusion, explainable Al, and a MERN-based monitoring layer. Python-based Al processing is intended to support model training and inference, while React, Node.js, Express, and MongoDB provide the application and monitoring components.
The framework is presented as a conceptual research direction and has not been experimentally validated in this review.

Future research should therefore focus on adaptive learning for evolving DNS threats, hybrid sequential and graph-based models, adversarially robust detection, privacy-preservinglearning, scalable edge or in-network processing, and operationally useful explainable Al. Experimental implementation and evaluation of the proposed framework willbenecessary to determine its effectiveness in terms of detection performance, cross-dataset generalization, robustness, inference latency, scalability, and practical security-operations support.

REFERENCES

Important: The list below contains the references whose bibliographic details I can currently verify directly from the uploaded PDFs. I am **not inventing the remaining entries.** The final numbering must be synchronized with the order in which the papers first appear in the manuscript.

[1] Javed, I. Rashid, S.Tahir, S. Saeed, A. M. Almuhaideb, and K. Alissa, "AdamW+: Machine learning framework to detect domain generation algorithms for malware," *IEEE* Access, vol. 12, 2024, doi: 10.1109/ACCESS.2024.3407546.

[2] R. Panigrahi, P. K. Sethy, S. K. Behera, M. Gupta, F. A. Alenizi, P. Suanpang, and A. Nanthaamornphong, "Analytical validation and integration of CIC-Bell-DNS-EXF-2021 dataset on security information and event management," *IEEE* Access, vol. 12, 2024, doi: 10.1109/ACCESS.2024.3409413.

[3] L. Hanintya, P. Sukarno, and A. A. Wardana, "Comparing explainable Al framework: Study case on detection of DNS exfiltration attack using neural network," *Procedia Computer Science,* vol. 269, pp. 1022-1032, 2025, doi: 10.1016/j.procs.2025.09.044.

[4] M. Anan, M. Nazzal, A. Khreishah, I. Khalil, N. Phan, and A. Sawalmeh, "STING: A stealthy backdoor attack on GNN-based malicious domain detection via DNS perturbations," *IEEE Open Journal of the Communications Society,* vol. 6, 2025, doi: 10.1109/0JCOMS.2025.3610784.

[5] M. Singh, M. Singh, and S. Kaur, "Tl-16 DNS labeled dataset for detecting botnets," *IEEE* Access, vol. 11, 2023, doi:10.1109/ACCESS.2023.3287141.

[6] K. Jerabek, K. Hynek, 0. Rysavy, and I. Burgetova, "DNS over HTTPS detection using standard flow telemetry," *IEEE* Access, vol. 11, 2023, doi: 10.1109/ACCESS.2023.3275744.

[7] L.A. C.Ahakonye, G. C. Amaizu, C. I. Nwakanma, J.M. Lee, and D.-S. Kim, "Classification and characterization of encoded traffic in SCADA network using hybrid deep learning scheme," *Journal of Communications and Networks,* vol. 26, no. 1, pp. 65-76, Feb. 2024.

[8] Arkadopoulou, **N.** Kostopoulos, **M.** Grammatikou, and V. Maglaris, "Feature engineering in 2 unsupervised DNS botnet detectors based on eXplainable Al," *IEEE* Access, vol. 14, 2026, doi: 10.1109/ACCESS.2026.3687021.

[9] Y. Baseri, M. Daghmehchi Firoozjaei, S. Sadeghi, A. Ghorbani, and W. Belanger, "Proactive cyber defense: User-centric risk assessment in DNS traffic through graph-based learning," *IEEE* Access, vol. 14, 2026, doi: 10.1109/ACCESS.2025.3648445.

[10] J B. Ali and G. Chen, "Neuromorphic quantum adversarial learning (NQAL): A bio-inspired paradigm for DNS over HTTPS threat detection," *EURASIP Journal on Information Security,* vol. 2025, Art. no. 35, 2025, doi: 10.1186/s13635-025-00215-5.

[11] M. Ochaba, M. Mrukowicz, J. Sarzynski, and W. Rzasa, "A scalable data acquisition system for the efficient processing of DNS network traffic," *Procedia Computer Science,* vol. 246, pp. 4702-4711, 2024, doi: 10.1016/j.procs.2024.09.335.

[12] J. Vanerio, C. Gyorgyi, and S. Schmid, "Poster: P4DME: DNS threat mitigation with P4 in-network machine learning offload," in *Proc. 6th European P4 Workshop (EuroP4 '23),* Paris, France, Dec. 2023, pp. 1-4, doi: 10.1145/3630047.3630251.

[13] K. Hu, H. Du, and Z. Wang, "DNScope: Detecting DNS misconfigurations through graph neural static reasoning," in *Proc. 2nd Workshop on Formal Methods Aided Network Operation (FMANO '25),* Coimbra, Portugal, Sep. 2025, pp. 1-6,doi: 10.1145/3750022.3750460.

[14] AdamW+: Machine Learning Framework to Detect Domain Generation Algorithms for Malware — Javed et al., 2024.

[15] Analytical Validation and Integration of CIC-Bell-DNS-EXF-2021 Dataset on Security Information and Event Management — Panigrahi et al., 2024.