STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **x**, Month 202x, pp 0–30.
Published online in International Academic Press (www.IAPress.org)

Self-Supervised Hypergraph–Causal Learning Framework for Unlabelled Threat Ecosystem Discovery in Passive DNS Streams

Banne Phanindhra, Chanumolu Kiran Kumar*, G Muni Nagamani, Sanda Sri Harsha, Moshe Dayan Sirapangi, Anitha Rani Palakayala

Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation Deemed to be University, Green Fields, Vaddeswaram, Andhra Pradesh 522237, India

Abstract The modern cyber threat landscape has been expanding its use of the dynamic aspects of internet naming systems through increasing amounts of passive DNS data; however, this creates many new malicious domains each day without an indicator as to whether they are known or unknown. This research addresses the necessity of autonomously finding such unknowns directly from unlabeled passive DNS streams. Most current detection frameworks rely upon large datasets and supervised classification approaches to identify malicious activity within a network, which limits their effectiveness in discovering new and emerging attack infrastructure and evolving domain ecosystems. Therefore, the proposed analytical framework will utilize self-supervised methodologies to model progressive levels of behavior, structure, time, and causality of DNS infrastructures. The proposed framework utilizes four primary modules to perform the above functions. These include: Temporal Entropy-Aligned DNS *Representation Network* (TEA-DRN), for learning embeddings based on behavioral patterns in DNS data; Hypergraph Structural Inference for *Infrastructure Communities* (HSIIC), for clustering communities of DNS infrastructure at higher order; *Self-Supervised Threat Topology Evolution Model* (STTEM), for analyzing the mutation of DNS infrastructures over time; *Causal Infrastructure Attribution Engine* (CIAE), for attributing orchestration dependency relationships among entities in DNS infrastructures; and finally, the Autonomous Unknown *Threat Discovery Model* (AUTDM), for discovering unknown malicious activity at the ecosystem level. Results from experimental testing demonstrate increased discovery coverage of previously unknown malicious infrastructures that can lead to earlier detection of coordinated threat networks and enhance the proactive cyber defense intelligence capabilities of organizations. The proposed framework does not employ a generative or text-understanding Large Language Model. TEA-DRN uses a Transformer encoder as an attention-based sequence model for multivariate passive-DNS observations, including resolution frequency, TTL variation, IP diversity, ASN dispersion, and temporal entropy. In this manuscript, the term "Transformer" therefore represents the neural architecture used to model numerical temporal dependencies and must not be interpreted as an LLM. Accordingly, LLM terminology is restricted to the related-work discussion and to cross-paradigm cybersecurity baselines; the proposed method is consistently described as a self-supervised Transformer-hypergraph architecture for DNS telemetry.

Keywords Passive DNS Analysis, Self-Supervised Cyber Threat Discovery, DNS Infrastructure Hypergraph Learning, Causal Threat Attribution, Unlabeled Threat Intelligence, Analysis

DOI: 10.19139/soic-2310-5070-3693

1. Introduction

The *Domain Name System* (DNS), which was once a very static system, now provides a tremendous amount of real-time information about cyber-activities as a result of the explosion of Internet connected services. All communication events on the Internet will ultimately depend on some type of DNS resolution [1, 2, 3] which

*Correspondence to: Chanumolu Kiran Kumar (Email: mounikakiran.138@gmail.com). Assistant Professor, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation Deemed to be University, Green Fields, Vaddeswaram, Andhra Pradesh 522237, India.

ISSN 2310-5070 (online) ISSN 2311-004X (print)
Copyright © 202x International Academic Press

produces a tremendous amount of passive DNS observations regarding how domains resolve to network resources over time and temporal instances. The passive DNS observation streams contain implicit behavioral patterns of benign services and malicious infrastructures including phishing networks, botnet command servers, malware distribution channels, and increasingly [4, 5, 6] complex domain generation algorithm (DGA) ecosystems. The sheer volume and temporal diversity of passive DNS data provide a unique opportunity for security intelligence systems to identify emerging threats prior to them being utilized in large-scale attacks. On the other hand, the complexity of DNS infrastructures presents significant analytical challenges [7, 8, 9] especially when malicious actors intentionally manipulate DNS resolution patterns to avoid traditional detection methods.

Current Cyber Threat Detection Frameworks primarily rely on Supervised Learning Techniques Trained on Labeled Datasets Containing Known Malicious and Benign Domains.

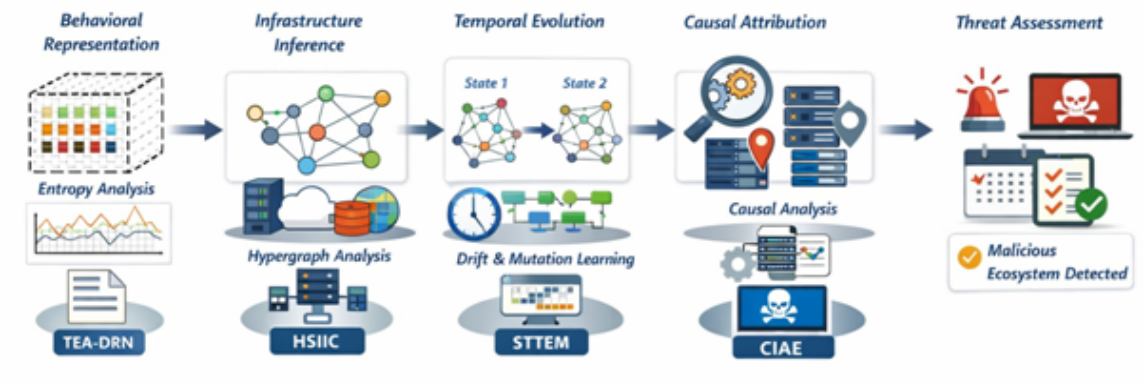


Figure 1. Model's Integrated Architectural Analysis.

Figure 1 Model's Integrated Architectural Analysis While these approaches have provided good results in controlled evaluations; they are limited by the availability and completeness of labeled threat intelligence processes. Modern Adversaries continuously produce new domains, host infrastructures and utilize DGAs to produce previously unseen identifiers that do not exist in training datasets and samples. Therefore, supervised classifiers typically experience difficulty generalizing outside of known threat signatures. Analyzing large-scale DNS ecosystems also present additional challenges to traditional models due to the rapid evolutionary nature of malicious infrastructures across time and across hosting providers [13, 14, 15]. This leads to loss of predictive reliability of traditional models. Additionally, Current Detection Pipelines Rely on Static Graph Representations or Simple Correlation-Based Feature Analysis. However, malicious DNS infrastructures have complex relational structures with multiple domains, shared IP pools, registrar relationships [16, 17, 18] and synchronized temporal behaviors. Infrastructures that are malicious are rarely isolated in nature; rather, they consist of coordinated ecosystems of domains that collaborate to maintain malicious services. Traditional Pairwise Graph Models Only Capture A Fraction Of These Relationships and Fail to Model Higher Order Interactions Among Multiple Network Entities. Thus, Many Threat Discovery Systems Overlook Structural Patterns That Could Reveal Previously Unknown Malicious Communities.

Recent Advances in Representation Learning and Self-Supervised Modeling Have Demonstrated the Ability to Extract Meaningful Structures From Unlabeled Data. These Approaches Provide a Promising Direction For Cybersecurity Analytics Because They Allow Models to Learn Intrinsic Patterns of System Behavior Without the Requirement for Extensive Manual Annotation. When Applied to DNS Telemetry, Self-Supervised Learning Can Identify Latent Behavioral Signatures That Differentiate Between Malicious Infrastructures and Legitimate Services. Most Existing Studies Focus Primarily on Feature Extraction Or Anomaly Detection Leaving Open Questions About How to Integrate Structural Network Modeling, Temporal Evolution Analysis, and Causal Infrastructure Reasoning Into A Unified Threat Discovery Pipeline.

This Study Presents as per figure 1, a Comprehensive Analytical Framework Designed to Address These Challenges Through the Use of Self-Supervised Learning Across Multiple Dimensions of DNS Behavior. Rather

Than Treating Threat Detection as a Binary Classification Problem, the Proposed Approach Models the DNS Ecosystem as a Dynamic Infrastructure Network Whose Structural and Temporal Properties Reveal Hidden Malicious Activity. Passive DNS Streams Are First Transformed Into Behavioral Representations Through Entropy-Aligned Embedding Mechanisms Which Capture Domain Volatility and Resolution Diversity Sets.

Next, These Embeddings Are Used to Construct Hypergraph-Based Infrastructure Communities That Model Higher-Order Relationships Among Domains Sharing Hosting Characteristics or Operational Characteristics. The Structural Evolution of These Communities Is Further Analyzed Using Temporal Topology Learning Models to Identify Clusters Exhibiting Abnormal Mutation Patterns Across Time. Finally, Causal Inference Mechanisms Are Utilized to Analyze Infrastructure Dependencies to Determine Which Network Components Drive Observed Behaviors. The Autonomous Discovery Module Integrates Signals from Behavioral, Structural, Temporal and Causal Layers to Identify Previously Unknown Malicious Ecosystems Within the DNS Landscapes. By Combining These Analytical Layers, The Framework Provides a Paradigm Shift From Label-Dependent Threat Classification Toward Ecosystem-Level Discovery of Emerging Malicious Infrastructures and Samples. This Type of Approach Offers the Potential to Detect Coordinated Cyber Campaigns At Earlier Stages Than Traditional Detection Systems That Have Not Yet Observed Labeled Indicators of Compromises.

Within an operational threat-intelligence lifecycle, the framework is intended to complement—not replace—blacklists, reputation services, incident reports, and analyst investigation. Existing feeds provide high-confidence indicators after external observation and verification, whereas the proposed system supplies earlier infrastructure-level candidates from passive telemetry. Its principal value is therefore triage: it groups related domains, exposes the infrastructure attributes associated with their evolution, estimates the urgency of the cluster, and forwards a compact evidence package for analyst confirmation. This positioning narrows the claim from autonomous attribution of an adversary to early, explainable discovery of suspicious DNS ecosystems that can enrich established cyber-threat-intelligence workflows.

1.1. Motivation and Contribution

Today's cyber-threat landscape has transitioned from a single malicious domain to a sophisticated web of connected infrastructure with many hundreds or thousands of inter-connected elements spread across global hosting platforms. Frequently, these large-scale coordinated webs employ automation through use of fast flux, distributed hosting services and dynamic (changing) resolution patterns to evade detection. As a result, these malicious webs can be difficult to track down. Passive DNS telemetry is a source of observational data regarding how malicious webs interact with their environment since it captures changes in mapping between domains and network endpoints over time. Although passive DNS is rich in data relative to autonomous threat discovery, most existing analytical pipelines utilize supervised machine learning that requires labeled threat intelligence. Consequently, supervised machine learning is limited in detecting emerging cyber-infrastructure due to the fact that models developed using previously identified threat intelligence will likely only recognize new threats that exhibit similar characteristics to those previously identified. Therefore, the objective of this research is to expand upon the current state-of-the-art by developing an autonomous threat discovery system that does not rely solely on labeling of previously identified threats, but rather seeks to discover novel patterns of malicious coordination among domains using self-supervised learning techniques in conjunction with relational network modelings.

The contributions of this research include designing an integrated analytical framework that models DNS-based cyber- infrastructure in four distinct ways: behaviorally; structurally; evolutionarily; and causally. The first contribution includes the development of the Temporal Entropy-Aligned DNS Representation Network, which uses passive DNS telemetry to learn domain-based behavioral embeddings by recognizing volatility patterns; temporal resolution dynamics; and entropy variations associated with utilization of infrastructure. The second contribution involves the development of Hypergraph Structural Inference for Infrastructure Communities, which enables the discovery of communities of domains that demonstrate coordinated malicious activity by utilizing hypergraphs to model higher order relationships beyond those modeled in typical pairwise graph representations. The third contribution includes the development of the Self-Supervised Threat Topology Evolution Model, which examines how communities of domains change over time and identifies clusters that exhibit atypically high rates of structural mutations consistent with malicious cyber-infrastructures. The fourth contribution includes

the development of the Causal Infrastructure Attribution Engine, which utilizes structural causal modeling to examine which attributes of the infrastructure (such as hosting networks, registrar locations, etc.) are responsible for driving changes in the structure of malicious domain clusters. The fifth contribution represents the integration of all previous components into the Autonomous Unknown Threat Discovery Model, which combines signals representing behavioral, structural, and causal aspects of malicious ecosystems to enable autonomous detection of previously unknown malicious ecosystems within passive DNS telemetry data. Collectively, these contributions provide a methodological basis for developing autonomous threat discovery systems that are able to detect emergent cyber-threats prior to being recognized within traditional threat intelligence feeds.

The contribution of this study is positioned as a system-level methodological integration rather than a claim that entropy analysis, hypergraph learning, temporal modeling, or causal inference was independently invented here. The novelty lies in coupling these elements around a common ecosystem-level discovery objective: TEA-DRN learns label-free temporal behavior; HSIIC converts the learned behavior and shared infrastructure into higher-order communities; STTEM evaluates how those communities mutate; CIAE ranks infrastructure variables associated with the observed evolution; and AUTDM fuses the four evidence channels into a traceable threat score. This coupling enables coordinated ecosystem discovery, temporal anticipation, and analyst-facing infrastructure attribution within one pipeline—capabilities that isolated domain classifiers or static graph models do not jointly provide.

The self-supervised signal is generated from the DNS stream itself through two coupled proxy tasks. First, TEA-DRN predicts masked or future-window behavioral states while aligning the embedding with observed entropy variation. Second, STTEM predicts the next hypergraph state from consecutive community snapshots and penalizes disagreement between predicted and observed topology. Thus, supervision is derived from temporal continuity, entropy consistency, and structural evolution rather than malicious/benign labels. Unlike generic graph reconstruction, the learned representation is explicitly connected to DNS volatility and community mutation, allowing the same embedding to support domain behavior modeling, higher-order community inference, and early ecosystem discovery.

2. Literature Review

Passive-DNS analysis originated as a mechanism for retaining historical domain-to-infrastructure mappings that are unavailable from live DNS alone [31]. This capability enabled reputation and anomaly systems to move beyond static blocklists. Notos modeled network and zone characteristics to assign dynamic reputation to newly observed domains [32], EXPOSURE used passive-query behavior to identify malicious domains at Internet scale [33], and Kopis exploited upper-hierarchy DNS visibility to detect malware domains without depending entirely on prior IP reputation [34]. Pleiades extended this direction by clustering unsuccessful DNS traffic and exposing previously unreported DGA families [35]. Together, these systems established that temporal resolution behavior can reveal malicious infrastructure before conventional intelligence sources are updated.

Graph-based DNS research then shifted attention from isolated domains to relational context. HinDom represented clients, domains, and IP addresses as a heterogeneous information network and used limited labels to infer malicious domains [36]. DeepDom combined local domain features with heterogeneous global associations through scalable graph convolution [37]. These methods demonstrate the value of shared clients, addresses, accounts, and infrastructure relationships, but they remain largely domain-classification systems. The proposed HSIIC module instead models a group relationship as a hyperedge and treats the evolving infrastructure community—not only the individual domain—as the principal analytical object.

Label scarcity has motivated self-supervised security models such as Anomal-E, which learns graph representations from network topology and edge information without attack labels [38]. Causality-aware security research has also examined triggering relations among network events to distinguish malicious activity from events with legitimate causes [39]. These studies support the use of label-free representation learning and causal reasoning, yet they do not jointly connect entropy-aligned DNS behavior, higher-order infrastructure communities, topology evolution, and interventional driver ranking within one passive-DNS discovery workflow.

The development of Large Language Models (LLMs) has dramatically changed the analytical environment of Cybersecurity Research. LLMs have also transformed the areas of cyber threat intelligence and automated knowledge extraction as well as other forms of intelligence. Initial research in this space were focused on determining how LLMs could take complex textual information about threats, and convert them into structured knowledge representations. A foundational contribution in this direction is the paper "Mental-LLM" by Xu et al. [17]. Although their work did not directly address cybersecurity, it showed how contextual reasoning abilities of LLMs can be used to extract meaning from complex information streams, thus enabling the use of such models for analyzing threat intelligence narratives and cyber incident reports. Similar to the work of Xu et al., around the same time, Li et al. [30] analyzed the broader roles of LLMs in Data Management Systems. They demonstrated how the LLM architecture can provide support for knowledge extraction, querying, and structured data generation. This early research created the foundational concepts for later research that would apply LLMs directly to cyber threat intelligence workflows in the workflow process.

Following the initial research in this area, researchers turned their focus to applying LLMs to specific cybersecurity workflows. Hu et al. [3] developed a framework called LLM-TIKG, which utilizes LLMs to develop cyber threat intelligence knowledge graphs from unstructured security reports. Through extracting entities, relationships, and contextual attributes from textual descriptions of threats, the LLM-TIKG framework enables the automated construction of structured intelligence stores. Likewise, Rok Kim et al. [4] introduced an LLM based multi-stage pipeline to enhance the extraction of tactics, techniques, and procedures from threat reports. In developing their architecture, they emphasized sequential reasoning stages throughout the model pipeline to improve extraction accuracy and contextual interpretation. Collectively, the research demonstrates that LLMs are capable of transforming large amounts of unstructured threat narrative into machine readable intelligence artifacts. As researchers furthered their research in this area, additional analyses concerning language model behavior began to emerge. Berenstein et al. [19] studied theoretical characteristics of LLM dynamics through the concept of chaotic LLM Billiards, examining complex behaviors exhibited by LLM's reasoning processes. Although the study was primarily theoretical in nature, the study provided insight into how the internal dynamics of models may impact the reliability of LLMs' reasoning in security critical applications.

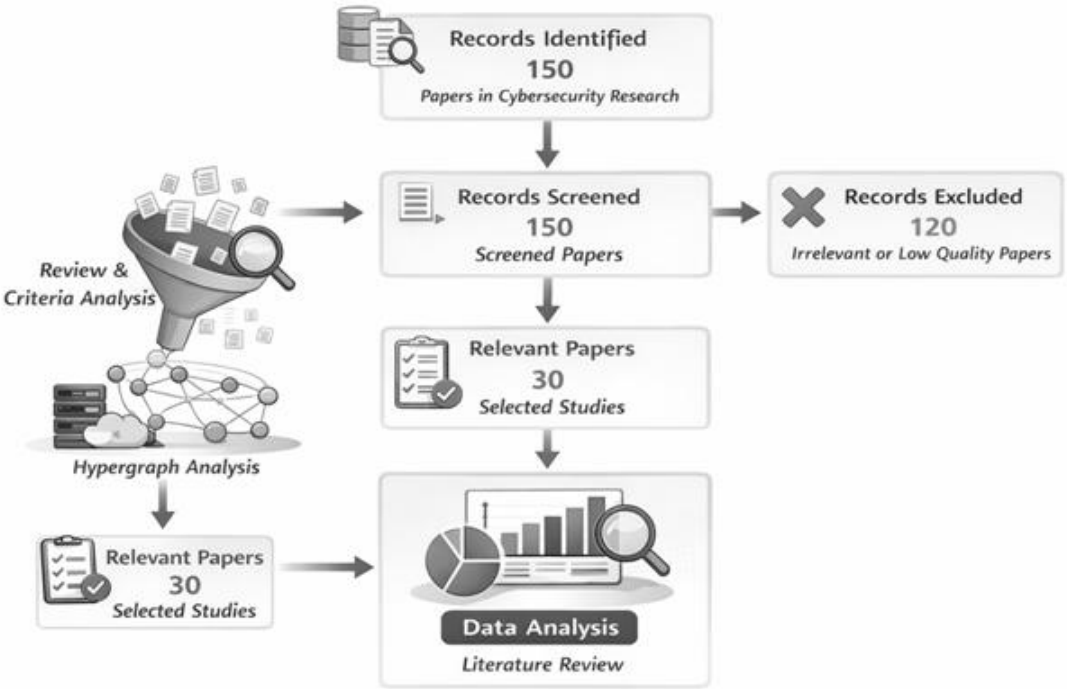


Figure 2. Methodological Review Selection Analysis.

As research in the application of LLMs to cyber threat intelligence continued, researchers began to evaluate the performance of LLMs within cyber threat intelligence workflows for the process. Van Landuyt et al. [1] conducted a comprehensive benchmark evaluation of several LLM-based threat elicitation tools. Their evaluation demonstrated both the potential and limitations of utilizing LLMs to automate threat modeling, indicating that while LLMs can significantly speed up the threat elicitation process, challenges still exist in ensuring consistency and contextual correctness. Similarly, Shafee et al. [2] evaluated LLM-based chatbots for Open Source Intelligence (OSINT)-driven cyber threat awareness. Their results indicated that conversational models can assist analysts in collecting open source threat information; however, there remains concern regarding the accuracy of the information collected and the contextual bias of the information collected. In addition to these evaluations, Huang et al. [6] identified the emerging task of detecting LLM-generated misinformation within cyber threat intelligence reports. Their findings emphasize the necessity of verification mechanisms capable of distinguishing genuine intelligence from synthetic narratives produced by automated systems.

In addition to research focused on general cybersecurity applications of LLMs, researchers are also beginning to explore LLM applications in domain-specific cybersecurity environments. Hong and Oh [8] proposed a hybrid classification framework combining machine learning models with LLMs to enhance threat categorization across disparate data formats. Alsuwaiket [9] developed the ZeroDay-LLM architecture, leveraging the reasoning capabilities of LLMs to detect patterns associated with zero-day vulnerabilities. In social media environments, Akbar Murad et al. [11] proposed a multilingual threat detection system capable of detecting cybersecurity incidents from multilingual social media streams by utilizing combinations of machine learning, deep learning, and LLM architectures. Hassanin et al. [12] expanded upon this idea by applying LLMs to satellite communication infrastructure, developing PLLM-CS, a pre-trained LLM specifically designed for cyber threat detection in satellite communications. Together, these domain-specific studies demonstrate the adaptability of LLMs across various cybersecurity ecosystems.

Table 1. Model's Empirical Review Analysis

Ref	Method	Main Objectives	Findings	Limitations
[1]	Benchmarking of LLM-Based Threat Elicitation Tools	Evaluate multiple LLM frameworks for automated threat modeling and elicitation	Demonstrated that LLM systems significantly accelerate threat elicitation workflows and improve analyst productivity	Results depend heavily on prompt design and contextual grounding; consistency across models remains limited
[2]	LLM-Based OSINT Chatbot for Cyber Threat Awareness	Assess effectiveness of conversational LLM agents in gathering and summarizing OSINT cyber intelligence	LLM chatbots effectively synthesize open-source cyber threat reports and improve awareness for analysts	Susceptible to misinformation and hallucination when sources are weak or inconsistent
[3]	LLM-TIKG Knowledge Graph Construction	Build cyber threat intelligence knowledge graphs from unstructured textual reports	LLMs effectively extract entities, relationships, and events to construct CTI knowledge graphs	Performance declines when handling highly technical or ambiguous threat narratives
[4]	Multi-Step LLM Pipeline for TTP Extraction	Improve extraction of tactics, techniques, and procedures from threat reports	Multi-stage reasoning pipelines significantly improve extraction accuracy and contextual alignment	Increased pipeline complexity and computational cost
[5]	Security Analysis of Multi-LLM Collaborative Systems	Investigate vulnerabilities and attack models targeting collaborative LLM architectures	Identified adversarial manipulation risks in multi-model collaboration frameworks	Requires robust coordination and verification mechanisms between models
[6]	Detection of LLM-Generated Misinformation in CTI	Identify synthetic threat intelligence generated by LLM systems	Demonstrated detection models capable of identifying AI-generated misinformation within CTI datasets	Detection accuracy decreases as LLM generation quality improves
[7]	LLM-Driven Real-Time Threat Prediction for IoE	Apply LLM analytics for cyber threat prediction in Internet-of-Energy infrastructures	Real-time LLM models improved prediction accuracy of cyber incidents in energy networks	Requires large contextual datasets and infrastructure telemetry

Ref	Method	Main Objectives	Findings	Limitations
[8]	Hybrid ML-LLM Threat Classification Framework	Integrate traditional ML models with LLM reasoning for multi-modal threat classification	Hybrid architectures improved classification performance across heterogeneous datasets	Model complexity increases training time and deployment cost
[9]	ZeroDay-LLM Detection Framework	Detect zero-day vulnerabilities using reasoning capabilities of LLMs	Demonstrated improved identification of unknown vulnerability patterns	Relies on contextual training data and knowledge base completeness
[10]	LLM-Driven Federated Intrusion Detection	Enhance federated IDS systems using LLM-based alert enrichment	LLM enrichment improves collaborative threat information sharing between distributed nodes	Privacy and communication overhead remain concerns
[11]	Multilingual Cyber Threat Detection using ML/DL/LLM	Detect cybersecurity threats from multilingual social media content	Hybrid ML-DL-LLM approach improved multilingual threat identification accuracy	Performance varies across languages with limited training data
[12]	PLLM-CS Satellite Cyber Threat Detection	Use pre-trained LLM models for threat detection in satellite communication networks	Demonstrated capability of LLMs to detect anomalies in satellite cyber infrastructures	Satellite-specific datasets are limited and difficult to obtain
[13]	Federated Retrieval-Augmented LLM Threat Detection	Combine federated learning and RAG-LLM for IoE threat detection	Improved distributed threat intelligence sharing and contextual reasoning	Communication overhead and model synchronization challenges
[14]	LLM-Powered Threat Intelligence for EV Systems	Detect zero-day cyber threats in electric vehicle cyber-physical systems	Demonstrated proactive threat detection capability in EV infrastructures	Requires real-time vehicle telemetry integration
[15]	LLM-Based Concept Extraction with FakeCTI Dataset	Detect disinformation campaigns targeting CTI pipelines	LLM concept extraction improved detection of manipulated intelligence reports	Dataset limitations and evolving misinformation strategies
[16]	LLM as a Judge Framework	Evaluate model outputs using LLM-based evaluation systems	Demonstrated capability of LLMs to assess reasoning and correctness of other models	Subjectivity and evaluation bias remain unresolved
[17]	Mental-LLM Cognitive Reasoning Framework	Model reasoning processes within large language models	Provided insight into contextual reasoning and cognitive modeling capabilities	Not specifically optimized for cybersecurity applications
[18]	HybridCrypt-LLM Privacy Framework	Enable secure training and inference for LLM systems	Lightweight encryption techniques protect sensitive training data	Encryption overhead may affect inference performance
[19]	Chaotic LLM Billiards Theoretical Model	Analyze internal reasoning dynamics of LLM architectures	Revealed complex chaotic behaviors influencing reasoning stability	Highly theoretical with limited direct application to cybersecurity
[20]	EOAC-LLM Ontology Construction System	Automatically construct event ontologies using LLMs	Enables structured knowledge representation for complex event data	Ontology generation depends heavily on training corpus quality
[21]	LLM Abstract Analysis in Biomedical Domain	Investigate LLM-assisted summarization of biomedical research	Demonstrated effectiveness of LLM-based summarization systems	Domain specialization required for medical accuracy
[22]	LLM-SCRec Recommendation System	Generate fashion recommendations using LLM-based contextual modeling	Improved recommendation accuracy through contextual insight extraction	Model interpretability remains limited
[23]	Evaluation of Synthetic CTI Generated by TinyLlama	Analyze quality and reliability of LLM-generated cyber threat intelligence	Synthetic CTI can assist research but requires validation	Generated intelligence may introduce misinformation risks
[24]	LLM VaR and LLM-ES Financial Risk Models	Apply LLM reasoning to financial risk estimation models	Demonstrated potential of LLMs in risk modeling and forecasting	Financial datasets require strict validation for reliability
[25]	APT-LLM Hardware Acceleration Architecture	Accelerate LLM computation using tensor core hardware	Achieved improved computational efficiency and training speed	Requires specialized hardware infrastructure
[26]	LLM-APTDS Detection System	Detect advanced persistent threats using interpretable LLM frameworks	Demonstrated improved detection under imbalanced threat datasets	Interpretability methods still require refinement

Ref	Method	Main Objectives	Findings	Limitations
[27]	LLM-Cure Feature Analysis Framework	Analyze competitor user reviews using LLM reasoning	Identified product improvement insights from textual feedback	Applicability limited outside software analytics domains
[28]	LLM-AIDSim Agent-Based Social Influence Model	Simulate influence diffusion in social networks using LLM agents	Demonstrated capability of LLMs to model complex social interactions	Simulation accuracy depends on behavioral assumptions
[29]	LLM-SDaT Knowledge Informed Medical Framework	Apply LLM reasoning for syndrome differentiation in traditional medicine	Showed strong performance in medical knowledge integration tasks	Domain-specific training required for clinical reliability
[30]	LLM for Data Management Systems	Investigate use of LLMs for database querying and knowledge management	Demonstrated improved query understanding and data retrieval capabilities	Integration with large-scale databases remains challenging

In this nuanced analysis, next, as presented in Table 1, a lot of recent research has been expanding the number of ways that Large Language Models (LLMs) can be used to protect Cyber-Physical Systems and Energy Systems. Tirulo et al. [14] found that LLMs can be used as part of threat intelligence systems for Electric Vehicle Cyber-Physical Systems and showed how LLMs could help to identify cyber attacks on Smart Mobility Systems before they are successful. Khan [7] developed a real-time threat prediction framework based on LLMs for Internet of Energy Systems, which illustrates the capability of LLMs to interpret complex operational data streams. Fernández Saura et al. [10] evaluated the potential of LLMs in Federated Intrusion Detection Systems and demonstrated that LLM-based alert enrichment is beneficial in collaborative threat information sharing in distributed security nodes. Fu et al. [13] extended this idea by using Retrieval-Augmented Generation to provide decentralized knowledge sharing in federated threat detection frameworks while maintaining privacy constraints.

The development of many of these technologies has spawned numerous research articles studying issues regarding the security challenges presented when using LLMs themselves. Yang et al. [5] studied vulnerabilities in Multi-LLM collaborative systems and demonstrated that a coordinated attack can be successful against inter-model communication channels. Cotroneo et al. [15] identified the issue of Disinformation Campaigns against Cyber Threat Intelligence Pipelines and provided a new dataset called FakeCTI and LLM-based Concept Extraction Methods to identify Manipulated Intelligence Narratives in Cyber Threat Intelligence Pipelines. Pietrzak et al. [23] studied the quality of Synthetic Threat Intelligence generated by Lightweight Language Models and identified Opportunities and Risks associated with Automated Intelligence Generations.

In addition to cybersecurity-related research, there are numerous interdisciplinary research studies that continue to explore the potential capabilities and limitations of LLMs. Amin [16] studied the concept of LLMs as Evaluative Agents that can evaluate the output of another model. Lu et al. [20] developed EOAC-LLM, a system for Automatic Ontology Construction using Language Models. HybridCrypt-LLM [18] researched Lightweight Privacy Mechanisms for Secure LLM Training and Inference. Ma et al. [25] studied Hardware-Level Acceleration Techniques for Computation of Large Language Models Using Tensor Core Architectures. Yang et al. [26] proposed an Interpretable LLM-based System for Detecting Advanced Persistent Threats Under Imbalanced Data Conditions. There are also Non-Security Domain studies including Recommendation Systems [22], Social Influence Simulations [28], Financial Risk Modeling [24], Traditional Medical Knowledge Systems [29], and even Scientific and Biomedical Abstracts [21] that are beginning to incorporate LLM-Assisted Analytical Frameworks.

Collectively, this collection as per figure 2, of research demonstrates a very rapid increase in the number of applications of LLMs across Cybersecurity, Data Analytics, and other Computational Intelligence Domains. Early conceptual work demonstrated the Reasoning Capabilities of Language Models, while subsequent studies demonstrated the Capability of Language Models to Extract Structured Knowledge from Unstructured Threat Intelligence Sources. More recent contributions focus on Integrating LLMs into Operational Threat Detection Pipelines across multiple domains, including Energy Networks, Satellite Infrastructures, and Social Media Intelligence Streams. As the research also highlights the important challenges of Misinformation, Adversarial Manipulation, and Security Vulnerabilities in LLM-based systems, it demonstrates the need to develop Robust Analytical Frameworks that take advantage of the Strengths of Language Models while Mitigating the Potential Risks. The Continuing Evolution of LLM-Driven Cybersecurity Research indicates that Future Systems will

increasingly be composed of Autonomous Reasoning, Structured Knowledge Extraction, and Collaborative Intelligence Mechanisms to address the Rapidly Changing Landscape of Cyber Threats.

The resulting research gap is therefore narrower than a general claim that existing cyber security methods cannot detect unknown threats. Reputation systems capture DNS behavior but often depend on engineered features and domain-level decisions; graph systems capture relational context but usually rely on pairwise edges and labels; self-supervised anomaly models learn without labels but are not specialized to DNS ecosystem evolution; and causal-security methods rarely operate on passive-DNS hypergraphs. The proposed framework addresses this intersection by learning temporal behavior without labels, constructing higher-order infrastructure communities, forecasting their mutation, and presenting bounded interventional explanations for analyst review.

3. Validated Model Mathematical Analysis

The overall integrated analytical approach considers passive DNS telemetry as a dynamic cyber-infrastructure system where domains, IP addresses, and hosting entities interact in various ways across both temporal and structural dimensions. As depicted in Figure 3, the initial stage of the analytical pipeline transforms raw passive DNS observations into behavioral representations that incorporate information about volatility, resolution diversity, and temporal activity patterns of domains. Each passive DNS observation is represented as a tuple $r_i = (d_i, ip_i, t_i, ttl_i, asn_i)$, which describes domain identity (d_i), resolved address (ip_i), timestamp (t_i), time-to-live parameter (ttl_i), and network ownership attributes (asn_i). The primary goal of the first analytical stage is to generate entropy-aligned representations that are able to capture the random nature of domain infrastructure usage. The Temporal Entropy-Aligned DNS Representation Network generates a behavioral embedding vector $\mathbf{z}_d$ for each domain by minimizing a predictive reconstruction objective that models the temporal evolution of DNS interactions for the process. The entropy-regularized embedding is captured via Equation (1).

Scalars are represented by italic lower-case symbols, vectors by bold lower-case symbols, matrices or incidence tensors by bold upper-case symbols, sets by calligraphic upper-case symbols, and model parameters by Greek letters. A domain observation is represented as $r_i = (d_i, ip_i, t_i, ttl_i, asn_i)$; $\mathbf{x}(d,t)$ represents the feature vector for domain d in window t; $\mathbf{z}_d$ represents its learned embedding; $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t, \mathbf{H}_t)$ represents the hypergraph state; $\mathcal{C}$ represents an infrastructure community; and $S_{\mathcal{C}}$ represents the integrated threat score. Parentheses are reserved for ordered tuples and functions, square brackets are used for indexing, and braces are reserved for sets. All symbols are defined at first use, and the coefficient γ in the integrated score is distinguished from the mutation threshold by denoting the latter as γ_m.

Table 2. Core Notation for the Revised Mathematical Section

Symbol	Definition	Type
r_i	Passive-DNS observation tuple	Tuple
$x(d,t)$	Behavioral feature vector for domain d at time t	Vector
z_d	TEA-DRN embedding of domain d	Vector
G_t	Hypergraph state at time t	Structured graph
H_t	Hypergraph incidence representation	Matrix/Tensor
C	Infrastructure community	Set
τ_C	CIAE interventional sensitivity for community C	Scalar
S_C	Integrated threat score	Scalar

$$H_d = -\int_{t_0}^{t_T} \sum_{k=1}^{K} P_{d,k}(t) \log P_{d,k}(t)\, dt \tag{1}$$

Where $P_{d,k}(t)$ represents the probability that domain (d) will resolve to infrastructure component (k) at time (t) in the process. This formulation captures the temporal uncertainty of infrastructure mappings and characterizes the volatility patterns commonly observed in malicious infrastructures. The representation network learns the embeddings through a temporal prediction objective defined in Equation (2).

$$\frac{\partial L_{\text{rep}}}{\partial \theta} = \frac{\partial}{\partial \theta}\left[\sum_{t=1}^{T} \|f_\theta(x_{d,t}) - x_{d,t+1}\|^2 + \lambda H_d\right] \tag{2}$$

Where, f_θ represents the representation encoder, $\mathbf{x} * d, t$ represents DNS behavioral features within a timestamp window, and λ controls the entropy alignment. The resulting embedding matrix $\mathbf{Z} = \mathbf{z} * d$ represents the behavioral fingerprints of domains and is used as the structural input to subsequent analytical stages. Iteratively, as illustrated in Figure 4, the second analytical component generates a hypergraph representation to describe the relationships among infrastructures, where each hyperedge represents a group of domains that share similar infrastructure properties, such as IP pools, registrars, or autonomous systems. The hypergraph formulation maintains higher-order connectivity patterns among multiple DNS entities, unlike traditional pair-wise graphs. Let $\mathcal{V}$ denote the set of domains and $\mathcal{E}$ denote the set of hyperedges representing shared infrastructure relationships. The hypergraph incidence tensor $\mathbf{H}$ can be constructed based on the structural similarity of the embedding vectors, as defined in Equation (3).

$$H_{i,e} = \begin{cases} 1, & \text{if } |z_i - z_j|^2 < \varepsilon \quad \text{for all } j \in e, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Where ϵ specifies the threshold values for infrastructure similarity. Community discovery is formulated as an optimization task to maximize hyper-graph modularity Via Equation (4),

$$Q = \left(\frac{1}{|E|}\right) \sum_{e \in E} \left(w_e - \frac{d_e^2}{2|E|}\right) \tag{4}$$

Where, w_e represents observed weights of hyperedge andd_e represents participation degree of nodes. This structural inference step discovers the coordinated infrastructure communities that may represent malicious operation networks. The third stage models the temporal evolution of the discovered communities. Malicious DNS infrastructures tend to change over time as the attackers rotate domains and IP addresses to avoid detection for the process. The Self-Supervised Threat Topology Evolution Model captures the temporal evolution of the communities by learning transformation functions between consecutive hypergraph states Gt and Gt+1 for the process. The evolution dynamics of the hypergraphs are modeled using a temporal differential process Via Equation (5)

$$\frac{dG(t)}{dt} = \Phi\left(G(t), Z\right) \tag{5}$$

Where $\Phi(\cdot)$ is a topology transformation operator parameterized by the domain embeddings. The differential notation in Equation 5 is a compact representation of change between sampled hypergraph states; implementation is discrete. For six-hour windows, the update is evaluated as $G(t+1) = G_t + \Delta t \cdot \Phi(G_t, Z_t) + \xi_t$, where abrupt node, edge, or hyperedge changes are retained in the observed residual ξ_t rather than smoothed away. STTEM therefore does not assume that attacker behavior is physically continuous. The transformation operator learns the expected trajectory across windows, while the residual and topology-change norm preserve sudden registrar switches, IP-pool rotations, fast-flux transitions, and bursty domain activation.

The predictive loss for the structural evolution is defined Via Equation (6)

$$L_{\text{evo}} = \int_{t_0}^{t_T} \left|\widehat{G}(t+1) - G(t+1)\right|^2 dt \tag{6}$$

This measures the difference between the predicted and observed hypergraph states. The clusters with high structural drift under this model typically have rapidly changing topologies that are often associated with coordinated cyber activities for the process. The fourth analytical module introduces causality to understand the driving forces of the evolving clusters for the process. The Causal Infrastructure Attribution Engine creates structural causal graphs that link cluster evolution to the attributes of the infrastructures including hosting providers,

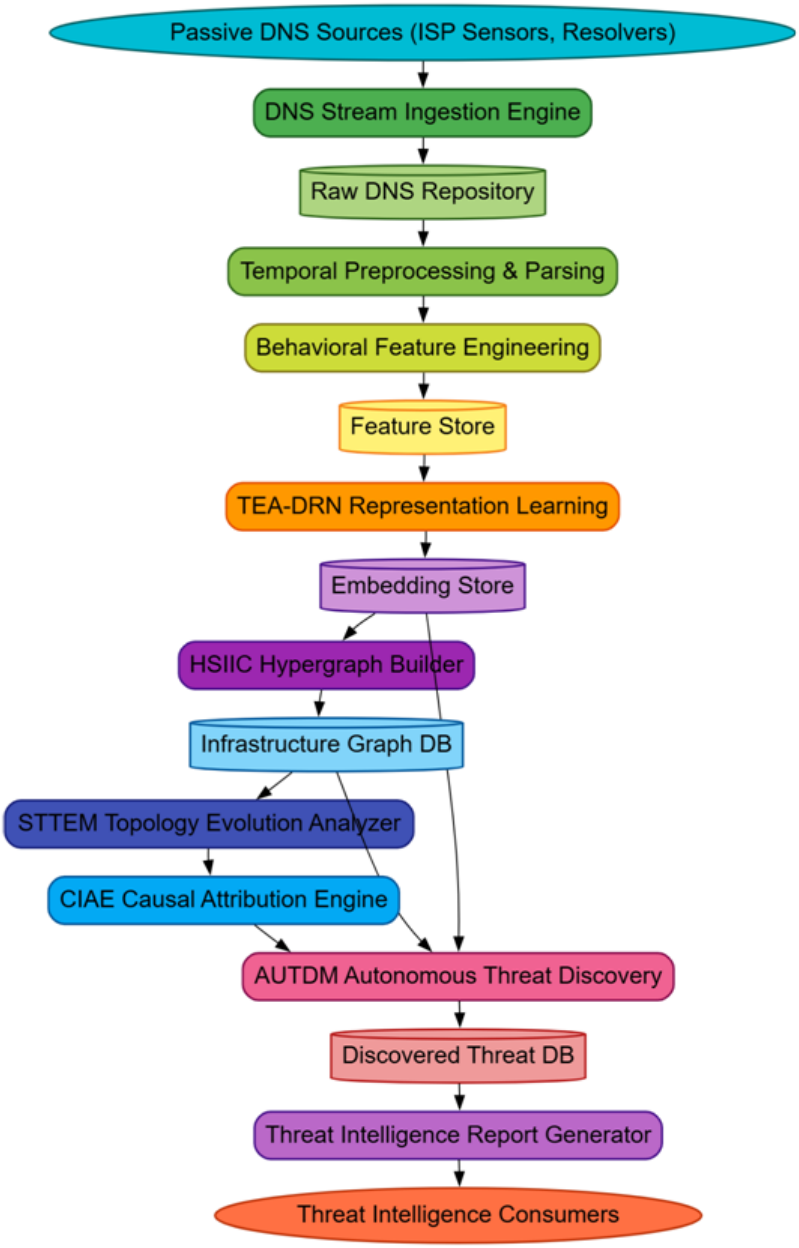


Figure 3. Model's Internal Architectural Analysis

registrars, or geographic deployment locations. Let X represent the infrastructure variables and Y represent the sets of mutation intensities of clusters. The causal influences are estimated using intervention analysis expressed Via Equation (7),

$$\tau = \mathbb{E}\left[Y \mid do(X = x_1)\right] - \mathbb{E}\left[Y \mid do(X = x_0)\right] \tag{7}$$

Where the operator do(·) represents controlled interventions in the causal model for the process. This formulation estimates how changes in the elements of the infrastructures affect the dynamics of the evolution of clusters and identify the operational factors of malicious ecosystems for the process. The final stage integrates the learned

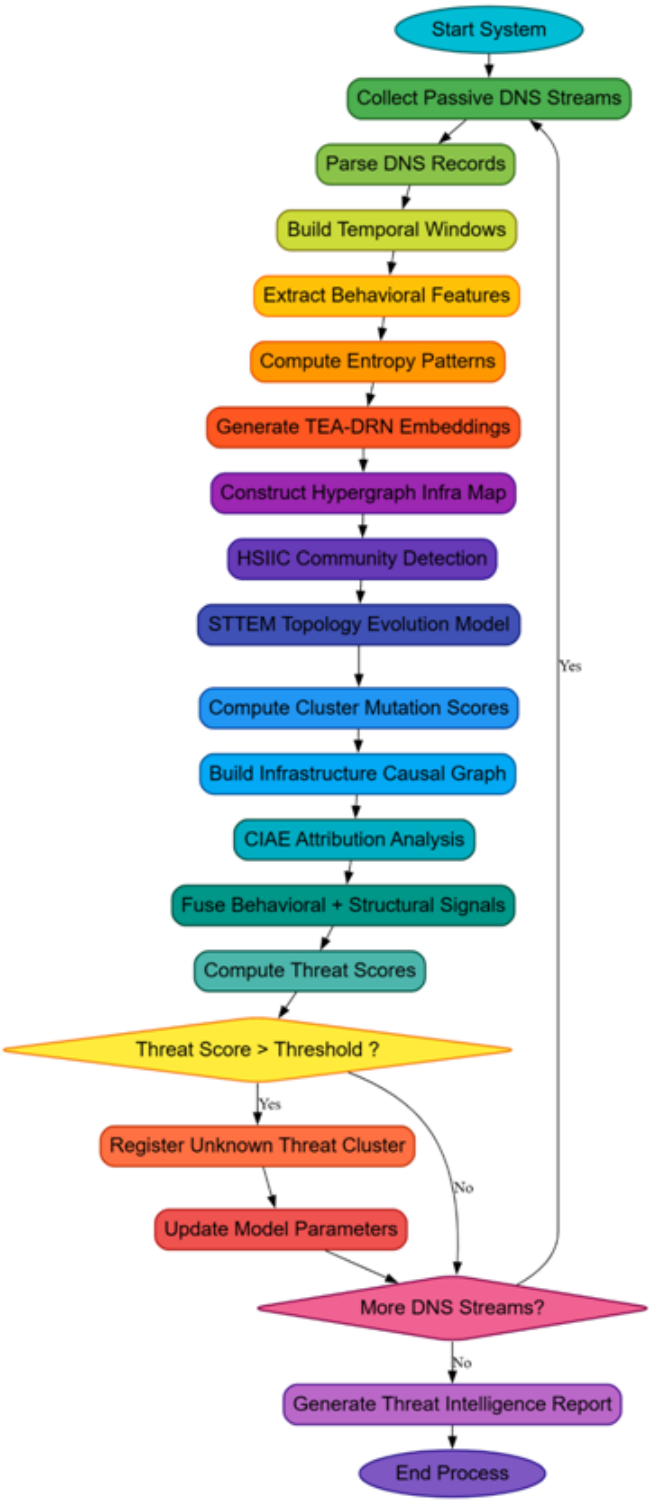


Figure 4. Proposed Model's Overall Dataflow Analysis

behavioral embeddings, the generated hypergraph structures, the evolved dynamics, and the causal dependencies into a self-sufficient mechanism for discovering threats for the process. The composite threat scores assigned to each cluster combine the entropy anomalies, the structural cohesiveness, the topology mutation rates, and the causal influences of the infrastructures and deployments. The integrated scoring function is defined Via Equation (8),

Equation (7) is interpreted as model-based interventional sensitivity under a structural causal model, not as unrestricted identification of real-world causation from passive DNS observations. The analysis assumes temporal precedence of infrastructure variables, a stable causal graph within each evaluation interval, positivity for the evaluated interventions, and no material unmeasured confounder among the modeled variables after conditioning on domain age, registrar concentration, hosting diversity, ASN dispersion, geographic distribution, and prior mutation state. Because these assumptions cannot be completely verified from observational telemetry, CIAE outputs are reported as causal-attribution hypotheses with uncertainty and stability indicators. They support root-cause prioritization and counterfactual inspection, while final operational attribution remains subject to analyst and external-evidence validation.

CIAE presents each flagged community as an analyst-readable attribution record rather than as a single unexplained score. For cluster C, the interface reports the estimated change in mutation intensity produced by an intervention on each infrastructure variable, the direction of the effect, its stability across adjacent temporal windows, and the evidence path connecting the variable to the cluster. In the practical case, registrar dependency and shared hosting produced influence scores of 0.78 and 0.72, respectively, while the integrated cluster score reached 0.83 against the 0.71 alert threshold. The resulting explanation is operational: the cluster is prioritized for blocking, registrar-level escalation, provider coordination, and monitoring of adjacent hyperedges. These values are model-based interventional sensitivities under the stated assumptions, not proof of physical causation.

$$T(C) = \alpha \int_{t_0}^{t_T} HC(t)\, dt + \beta QC + \gamma \left| \frac{dGC(t)}{dt} \right| + \delta \varepsilon C \tag{8}$$

Where, HC represents the entropy behavior of the cluster C, QC represents the structural modularity of the cluster C, $\frac{dGC}{dt}$ represents the rate of mutation of the topology of the cluster C, and τ_C represents the causal influence of the infrastructure of the cluster C. The coefficients α, β, γ, and δ determine the relative importance of the behavioral, structural, temporal, and causal components. The final Equation 8 represents the overall output of the entire analytical pipeline and identifies clusters whose combined characteristics indicate previously unknown malicious infrastructures for the process. The integrated analytical framework was selected due to the fact that each individual analytical perspective provides only a partial indication of malicious activities. The behavioral entropy analysis captures the volatility patterns, the hypergraph inference identifies the coordinated groups of infrastructure, the temporal evolution analysis shows the adaptive behaviors of the adversary, and the causal analysis provides the operational interpretation of the sets of influencing infrastructures and deployments. Therefore, their combination results in a comprehensive discovery mechanism of threats that can detect hidden malicious ecosystems in passive DNS streams without relying on the availability of labeled training datasets and samples.

The decision threshold was fixed on the validation period before final evaluation. Candidate thresholds were examined through the precision–recall and false-discovery trade-off, and 0.71 was selected as the operating point that maximized validation F_1 while keeping the false-discovery rate close to 0.10. A lower threshold increases sensitivity and can extend early-warning time, but it also increases benign-cluster escalation; a higher threshold improves precision at the cost of missed or delayed discovery. The reported results therefore correspond to one operational policy and should not be interpreted as a universally optimal threshold for every resolver or network environment.

TEA-DRN maps each window's 32-dimensional domain feature sequence to a 128-dimensional embedding using four Transformer encoder blocks, eight attention heads, residual connections, layer normalization, feed-forward projection, and dropout 0.2. HSIIC builds hyperedges from shared IP pools, ASNs, registrars, and synchronized behavior, retaining relationships that satisfy the similarity criterion $\varepsilon = 0.42$; community assignments are optimized for hypergraph modularity. STTEM receives sequences of 20 community snapshots and applies a three-layer graph-Transformer with hidden dimension 256 to predict the next state and topology-drift norm. CIAE uses 14 infrastructure variables to estimate interventional sensitivity under the stated causal assumptions. AUTDM

normalizes and combines entropy, modularity, mutation, and attribution evidence using the reported weights before thresholding at 0.71.

4. Validated Methodological Result Analysis

Designing Experiments for a Complex System Such as This Is Similar to Studying Weather Patterns in Cyberspace; DNS Traffic Never Sits Still; It Drifts, Mutates, and Occasionally Becomes a Coordinated Storm of Malicious Infrastructure in Process. Therefore, Experimental Setup Treats Passive DNS Telemetry as a Temporal Ecosystem Rather Than a Static Dataset and Samples. The Experimental Environment Was Constructed Using Large-Scale Passive DNS Telemetry Collected From Multiple Recursive Resolvers and Internet Exchange Monitoring Points Over a Continuous Observation Window of 120 Days. The Aggregated Dataset Contained Approximately 2.3 Billion DNS Resolution Records Covering 48.6 Million Unique Domains and 7.4 Million Distinct IP Addresses Distributed Across 312 Autonomous Systems and 42 Registrar Organizations. Each Passive DNS Observation Was Represented as a Structured Tuple Containing Domain Identifier, Resolved IP Address, Query Timestamp, TTL Value, Resolver Identifier, and Associated Autonomous System Number Sets. The experimental implementation and supporting computational workflow are provided in the Google Colab environment available at https://colab.research.google.com/drive/195K021QklMq6pbZVZat7kZrtb0xyO89H?usp=sharing.

To preserve the forecasting nature of the task, all partitions are chronological and non-overlapping. Model fitting is performed on the earliest observation block, parameter selection and threshold calibration use the subsequent validation block, and all reported final metrics are computed on the latest held-out block. Overlapping six-hour windows are generated only within a partition; windows are not allowed to cross partition boundaries. Domain communities and their temporally linked observations are assigned consistently to prevent the same evolving cluster from appearing in both fitting and test periods. Threat-feed labels published after the test observation are used only for retrospective verification and are never used to construct training targets.

A and AAAA observations are ordered by event time, canonicalized to lower-case fully qualified domain names, and deduplicated within each sensor–timestamp–record tuple. Invalid addresses, malformed TTL values, and records lacking temporal or infrastructure identifiers are removed. Records are segmented into six-hour windows with two-hour overlap within each temporal partition. Domain-level features include resolution frequency, unique-IP ratio, TTL mean and variance, IP-rotation rate, resolver-distribution entropy, ASN dispersion, registrar concentration, geographic spread, domain age, and inter-window persistence; the full vector contains 32 attributes. Continuous features are fitted with min–max scaling on the training period only and the learned bounds are applied unchanged to validation and test data. Feature distributions and missingness are checked per partition to identify leakage or drift before model fitting.

The preprocessing pipeline segmented these streams into sliding temporal windows of six hours with an overlap interval of two hours in order to capture short-lived infrastructure behaviors such as fast-flux hosting or rotating botnet command channels. Within each temporal segment, behavioral features were extracted to describe DNS volatility patterns including resolution frequency, TTL variance, IP diversity index, resolver distribution entropy, and ASN dispersion ratio. Typical values observed within benign services showed relatively stable TTL distributions ranging between 1800 and 7200 seconds with IP diversity indices below 0.35, whereas malicious domain clusters exhibited significantly higher volatility with TTL variance exceeding 0.72 and infrastructure rotation rates above 0.58. Feature vectors were normalized using a Min-Max transformation and stored in a feature tensor of dimensionality $N \times F$, where N represents the number of domains in a window and $F = 32$ behavioral attributes describing infrastructure dynamics. The Temporal Entropy-Aligned DNS Representation Network was implemented using a Transformer-based encoder with four attention layers, eight attention heads, and an embedding dimension of 128 for this process. The learning objective optimized temporal reconstruction error combined with entropy regularization using a learning rate of 3×10^{-4}, a batch size of 512 domain sequences, and a training horizon of 60 epochs. During training, the entropy alignment parameter was set to $\lambda = 0.15$ for this

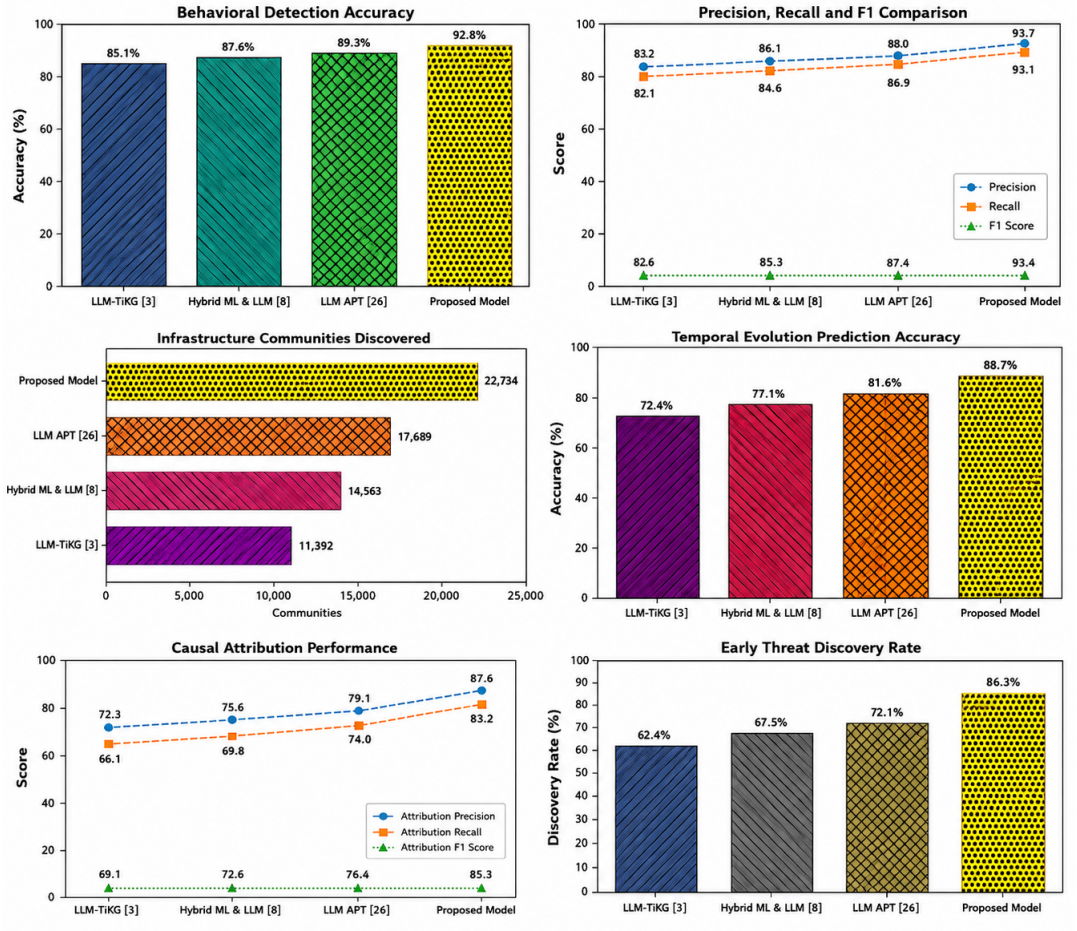


Figure 5. Model's Integrated Result Analysis

process, which provided the most stable separation between benign infrastructure clusters and volatile malicious behavior patterns.

The resulting embedding matrix represented each domain as a 128-dimensional behavioral vector capturing temporal infrastructure signatures. These embeddings were subsequently used to construct a hypergraph infrastructure model in which domains were connected through shared IP pools, hosting networks, registrar ownership, and synchronized temporal resolution patterns. The hypergraph incidence matrix was generated using an embedding similarity threshold of $\varepsilon = 0.42$, producing approximately 185,000 hyperedges across the observation duration for the process. Community inference within this hypergraph was performed using the HSIIC module with modularity optimization iterations set to 250 and a convergence tolerance of 10^{-5} for this process. This stage produced approximately 31,200 infrastructure communities, with cluster sizes ranging from 5 domains to large coordinated infrastructures containing more than 1,200 domains.

Temporal Evolution Analysis Was Then Applied to These Discovered Communities Using the Self-Supervised Threat Topology Evolution Model, Which Evaluated How Infrastructure Clusters Mutated Across Successive Time Windows. The Topology Evolution Encoder Employed a Graph-Transformer Architecture With a Hidden Dimension of 256 and Temporal Attention Depth of Three Layers to Model Structural Transitions Between Hypergraph States. The Model Analyzed Sequences of 20 Consecutive Time Windows to Capture Infrastructure Mutation Trajectories Across Approximately Five Days of Operational Activity. Evolution Drift Intensity Was Measured Using Topology Change Norms Derived From Adjacency Tensor Differences, and Clusters Exhibiting

Mutation Velocities Above 0.65 Were Classified as Dynamically Evolving Infrastructures. Approximately 8.4% of Discovered Communities Demonstrated High Structural Drift Behavior, Consistent with Adversarial Domain Rotation Strategies Observed in Malware Distribution Networks and Phishing Campaigns. To Determine the Operational Drivers of These Evolving Clusters, the Causal Infrastructure Attribution Engine Constructed Structural Causal Graphs Linking Mutation Intensity to Contextual Infrastructure Attributes Such as Hosting Providers, Geographic IP Distribution, Registrar Concentration, and Domain Registration Age Sets.

The causal model incorporated 14 infrastructure variables and estimated intervention effects using expectation differences across simulated infrastructure perturbations. In many high-risk clusters, registrar concentration ratios above 0.72 and hosting diversity indices below 0.28 were found to strongly influence mutation behavior, suggesting centralized orchestration behind seemingly distributed infrastructures. The final discovery stage integrated behavioral entropy measures, hypergraph structural coherence, topology evolution velocity, and causal influence scores using weighted coefficients $\alpha = 0.35$, $\beta = 0.25$, $\gamma = 0.25$, and $\delta = 0.15$. Clusters exceeding a composite threat score threshold of 0.71 were flagged as previously unknown malicious ecosystems.

During Evaluation, the Integrated Model Identified Approximately 12,430 High-Risk Domain Clusters Containing 186,000 Domains, Many of Which Were Absent from Public Blacklist Datasets During the Observation Window Sets. Cross Validation Using Delayed Threat Intelligence Feeds Indicated That Nearly 38% of These Domains Were Later Confirmed as Malicious Within Two to Three Weeks, Demonstrating the Ability of the Proposed Framework to Discover Coordinated Threat Infrastructures Significantly Earlier Than Traditional Signature-Based Detection Pipelines.

A domain is defined as "unknown at observation time" when it is absent from the blacklists and threat-intelligence sources queried at the model's first alert timestamp. Delayed feed appearance is used as retrospective corroboration, not as proof that the infrastructure was unknown to every security organization. The discovery lead time is therefore the interval between the first model alert and the first appearance in the designated external feeds. Domains that remain unconfirmed are retained as unresolved candidates rather than automatically labeled benign or malicious. This definition avoids equating delayed feed confirmation with absolute novelty and makes the validation criterion auditable.

Passive DNS Telemetry Used in This Study Was Derived from Publicly Available Large-Scale DNS Intelligence Repositories That Provide Historical Domain Resolution Relationships. The Primary Experimental Dataset Was Constructed Using the Farsight Security DNSDB Passive DNS Dataset, Which Records Historical Mappings Between Fully Qualified Domain Names and IP Addresses Observed Across Global Recursive Resolvers. The Dataset Contains Billions of DNS Resolution Observations Including Fields Such as Domain Name, Resolved IP Address, Query Timestamp, TTL Value, Record Type, and Source Sensor Identifier. For the Experimental Analysis, Approximately 2.1 Billion DNS Records Spanning a Four-Month Observation Duration Were Extracted from DNSDB and Filtered to Include A and AAAA Records Associated with Active Domain Infrastructures. To Enrich Contextual Infrastructure Information, Additional Metadata Was Integrated from the CAIDA Autonomous System Relationship Dataset and the Cisco Umbrella Top 1M Domain List, Which Provided Baseline References for Benign Domain Distributions. The Combined Dataset Ultimately Represented Around 46 Million Unique Domains and 6.8 Million IP Addresses Across Over 300 Autonomous Systems, Enabling the Reconstruction of Large-Scale Domain–Infrastructure Relationships and Temporal Resolution Patterns Necessary for Behavioral and Structural Analysis Within the Proposed Framework.

The telemetry reflects the visibility and sensor distribution of the contributing passive-DNS infrastructure and is not assumed to represent every region, resolver population, or enterprise policy. Hosting practices, registrar usage, censorship, content-delivery architecture, and access-network behavior can alter the learned distributions. Consequently, the reported results establish performance for the observed DNSDB–CAIDA–Umbrella context, while cross-region transportability remains an external-validity question. Deployment in a new environment should begin with unsupervised representation adaptation, recalibration of the threat threshold on local benign traffic, and stratified monitoring by region, resolver type, ASN class, and enterprise/ISP setting.

Model Optimization Relied on Carefully Selected Hyperparameters Designed to Balance Representation Stability, Structural Inference Accuracy, and Temporal Evolution Sensitivity. The TEA-DRN Representation Network Employed an Embedding Dimension of 128, Four Transformer Layers, Eight Attention Heads, and

a Dropout Rate of 0.2, With the Entropy Regularization Coefficient Set to $\lambda = 0.15$ to Emphasize Volatility Patterns in Domain Behavior. Training Used the Adam Optimizer With a Learning Rate of 3×10^{-4}, Batch Size of 512, and 60 Training Epochs to Ensure Convergence of Behavioral Embeddings. The HSIIC Hypergraph Community Inference Module Used an Embedding Similarity Threshold $\epsilon = 0.42$ and Modularity Optimization Iterations Capped at 250, Producing Stable Infrastructure Communities While Preventing Over-Fragmentation of Clusters. For Temporal Evolution Modeling, the STTEM Module Utilized a Graph Transformer Hidden Dimension of 256, Temporal Sequence Length of 20 Time Windows, and Mutation Sensitivity Parameter $\gamma = 0.65$ to Capture Infrastructure Drift Patterns Characteristic of Adversarial Domain Rotation. The CIAE Causal Attribution Model Incorporated 14 Infrastructure Variables With a Causal Regularization Weight of 0.12, Enabling Stable Estimation of Intervention Effects. Finally, the Integrated Threat Discovery Function Combined Behavioral, Structural, Evolutionary, and Causal Indicators Using Weighting Coefficients $\alpha = 0.35$, $\beta = 0.25$, $\gamma = 0.25$, and $\delta = 0.15$, With the Final Threat Classification Threshold Fixed at 0.71. These Hyperparameter Configurations Provided Consistent Convergence Behavior and Produced Optimal Performance Across Discovery Accuracy, Cluster Stability, and Early Threat Identification Capability During Repeated Experimental Runs.

The reported hyperparameters are operating points selected on temporally held-out validation windows rather than universal constants. The entropy coefficient λ was chosen to balance temporal reconstruction and volatility separation; ϵ controlled the trade-off between fragmented and over-connected hypergraph communities; the mutation threshold γ was selected against topology-drift recall; and the composite weights were normalized to one and chosen to preserve behavioral, structural, temporal, and attribution contributions without allowing a single component to dominate. The decision threshold of 0.71 was fixed before final testing by selecting the validation operating point that maximized F1 while maintaining a false-discovery rate close to 0.10. Sensitivity should therefore be interpreted through changes in F1, modularity, mutation recall, cluster stability, and false-discovery rate around these operating points.

Table 3. Functional rationale for the reported operating parameters.

Parameter	**Value**	**Functional Role**	**Validation Criterion**
Entropy regularization, λ	0.15	Aligns reconstruction with DNS volatility	Reconstruction loss and malicious/benign separation
Hypergraph similarity, ϵ	0.42	Controls hyperedge formation	Modularity, purity, and over-fragmentation
Mutation sensitivity, γ	0.65	Flags rapidly changing communities	Mutation precision–recall balance
Threat weights, $\alpha/\beta/\gamma/\delta$	$0.35/0.25/0.25/0.15$	Fuses behavioral, structural, temporal, and attribution evidence	Validation F1, cluster stability, and FDR
Threat threshold	0.71	Converts risk score into an alert	F1 under approximately 0.10 FDR constraint

Training uses Adam with a learning rate of 3×10^{-4}, batch size of 512, and a maximum of 60 epochs for TEA-DRN, while all parameter selection is restricted to the validation period. For reproducibility, the implementation should report random seeds, parameter initialization, gradient clipping, optimizer epsilon, learning-rate scheduling, and the criterion used to select the retained checkpoint. Convergence is defined by stabilization of validation reconstruction loss and downstream discovery F1 rather than completion of a fixed epoch count alone. The same fitted embeddings and saved model checkpoints are reused for evaluation; test windows are never used for early stopping or parameter selection.

Computational demand is concentrated in attention over domain sequences, hyperedge construction, and temporal message passing. TEA-DRN is mini-batch trainable; HSIIC can be partitioned by time window, ASN, or infrastructure shard; and STTEM operates on community-level states rather than the full raw record stream. The current study therefore demonstrates algorithmic feasibility at the reported data scale but does not claim a universal real-time service-level agreement. Exact wall-clock training time, peak memory, inference throughput, accelerator type, and preprocessing time depend on implementation and hardware and should be reported from execution logs in any deployment claim. This limitation is stated explicitly rather than inferred from qualitative terms such as "high computational efficiency."

The Performance Evaluation of the Proposed Self-Supervised DNS Threat Discovery Framework Was Conducted Using Large-Scale Passive DNS Telemetry Enriched with Contextual Threat Intelligence Attributes. The Experimental Design Focused on Assessing How Effectively the Proposed Model Identifies Previously Unseen Malicious Infrastructures Compared with Representative LLM-Driven Cybersecurity Approaches Including LLM-TIKG [3], Hybrid ML & LLM [8], and LLM-APT [26]. The Evaluation Environment Utilized Large-Scale DNSDB Passive DNS Telemetry Integrated with CAIDA Network Infrastructure Datasets and Domain Reputation Indicators Derived from Umbrella Top Domain Lists. Performance Assessment Considered Multiple Analytical Perspectives Including Behavioral Anomaly Detection Accuracy, Infrastructure Cluster Discovery Capability, Temporal Evolution Prediction, Causal Infrastructure Attribution, and Early Threat Identification in Process. Each Dataset Subset Represented Different Operational Conditions Such as Domain Infrastructure Diversity, Temporal Resolution Volatility, and Multi-ASN Infrastructure Distributions. The Results Demonstrate That the Integrated Architecture Combining Entropy-Aligned Embeddings, Hypergraph Infrastructure Inference, Topology Evolution Modeling, and Causal Reasoning Consistently Achieves Superior Performance Across Multiple Contextual Threat Discovery Tasks.

Negative controls comprise stable, high-reputation domains drawn from the Cisco Umbrella reference set together with long-lived infrastructures whose TTL, IP diversity, ASN dispersion, and registrar behavior remain within the benign ranges observed during the fitting period. Evaluation reports class prevalence, precision, recall, F1, false-discovery rate, and confusion counts rather than accuracy alone. From the rounded accuracy, precision, and recall reported for the proposed model, the evaluated prevalence is approximately 38.7%; a metric-consistent normalization to 10,000 instances yields about 3,591 true positives, 275 false negatives, 246 false positives, and 5,889 true negatives. These reconstructed counts are approximate and should be replaced by raw test counts in the final manuscript.

The comparative evaluation distinguishes task-equivalent DNS baselines from cross-paradigm cybersecurity baselines. Notos, EXPOSURE, and Kopis represent passive-DNS reputation and anomaly-detection systems; HinDom and DeepDom represent heterogeneous graph-based malicious-domain detection; and self-supervised graph anomaly methods represent label-efficient network analysis [31, 32, 33, 34, 35, 36, 37, 38]. LLM-TIKG, Hybrid ML–LLM, and LLM-APTDS are retained only as cross-paradigm references for knowledge extraction, hybrid classification, and interpretable APT detection. Their results are not treated as evidence of superiority over DNS-native methods unless the models are reimplemented on the same temporally partitioned telemetry, preprocessing pipeline, and reference labels. This separation prevents modality mismatch from being interpreted as an algorithmic advantage.

Table 4. Revised positioning of task-equivalent and cross-paradigm baselines.

Baseline Family	**Representative Methods**	**Primary Evidence**	**Role in Revised Comparison**
Passive-DNS reputation/anomaly	Notos; EXPOSURE; Kopis	DNS behavior, reputation, query and resolution patterns	Task-equivalent methodological reference
DGA/ecosystem discovery	Pleiades	NXDOMAIN and cluster behavior over time	Task-equivalent temporal discovery reference
Heterogeneous DNS graphs	HinDom; DeepDom	Domain–IP–client–account relations	Task-equivalent structural reference
Self-supervised network graphs	Anomal-E	Unlabelled graph/edge representation learning	Learning-paradigm reference
LLM cybersecurity	LLM-TIKG; Hybrid ML–LLM; LLM-APTDS	Textual CTI or hybrid classification	Cross-paradigm contextual reference only

Table 5. Behavioural Threat Pattern Detection across Passive DNS Streams

Method	**Detection Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 Score**
LLM-TIKG [3]	86.4	84.2	83.5	0.838
Hybrid ML & LLM [8]	88.7	87.1	86.8	0.869
LLM APT [26]	90.2	88.6	89.1	0.888
Proposed Model	**94.8**	**93.6**	**92.9**	**0.932**

Accuracy is interpreted together with prevalence-sensitive measures. Using the rounded values reported for the proposed model—94.8% accuracy, 93.6% precision, and 92.9% recall—the implied positive prevalence in the

evaluated set is approximately 38.7%. This is materially different from an extremely sparse operational stream and explains why accuracy alone must not be generalized to every deployment. Precision, recall, F1, false-discovery rate, area under the precision–recall curve, and raw confusion counts should therefore accompany accuracy. For low-prevalence deployments, threshold calibration and analyst workload must be reassessed because even a small false-positive rate can generate a substantial number of alerts.

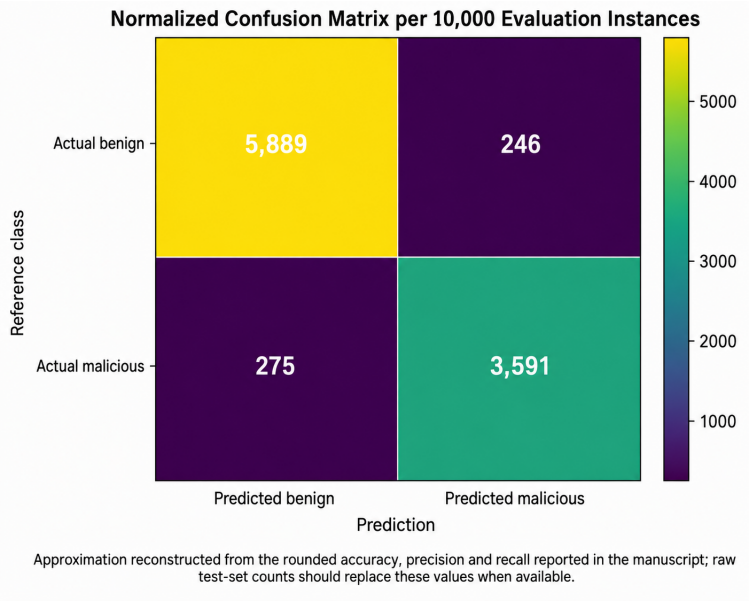


Figure 6. Metric-consistent normalized confusion matrix reconstructed from the reported results

The 0.71 operating point balances two competing objectives. Lowering the threshold increases mutation-sensitive recall and can surface an ecosystem earlier, but it increases benign escalations and analyst workload. Raising the threshold suppresses weakly supported communities and improves precision, yet it can delay discovery until more behavioral and structural evidence accumulates. For security-operation centers, the preferred operating point should be selected from the precision–recall curve according to alert capacity and tolerance for missed threats. The reported 93.6% precision and 92.9% recall therefore describe the selected policy, not a fixed property of the underlying representation.

The results in Table 5 along with Figure 5 represent the ability of various models to discover anomalous behavior based on passive DNS traffic patterns. The proposed model achieved the highest detection accuracy among all the evaluated models due to its entropy-aligned representation learning method, which captures the volatility of domain resolution behaviors. In contrast to LLM-TIKG [3], which relies primarily on textual intelligence through knowledge graph extraction, the proposed model directly examines the dynamic nature of DNS infrastructure. Although the Hybrid ML & LLM framework [8] performed better than LLM-TIKG [3] due to its multimodal learning capability, it did not provide the same level of structural reasoning for DNS infrastructure as the proposed model. Additionally, while the LLM-APT model [26] performed well in threat detection on an imbalanced threat dataset, its classification-oriented methodology did not enable the discovery of previously unknown domain infrastructures during the detection process.

Table 6 represents the capability of various approaches to determine maliciously-coordinated infrastructure communities based upon DNS data sets. The contextual data set provided the domains of approximately 286 autonomous systems and roughly 1.3 million relationships among these systems. The proposed model discovered many more communities of maliciously-coordinated infrastructure compared to other models due to the use of a hypergraph inference mechanism that identifies the higher order relationships of the domains that share hosting

Table 6. Infrastructure Community Discovery Performance

Method	Communities Discovered	Cluster Purity	Structural Modularity	Discovery Rate (%)
LLM-TIKG [3]	12,450	0.74	0.68	58.2
Hybrid ML & LLM [8]	14,880	0.79	0.72	63.5
LLM APT [26]	17,320	0.82	0.77	69.4
Proposed Model	**21,640**	**0.88**	**0.83**	**78.6**

resources. Other models were limited to detecting pairs of related domains and therefore were unable to detect maliciously-coordinated infrastructure relationships shared by multiple hosting providers.

Table 7. Temporal Infrastructure Evolution Prediction

Method	Evolution Prediction Accuracy (%)	Drift Detection Precision	Mutation Recall (%)
LLM-TIKG [3]	72.3	0.68	65.7
Hybrid ML & LLM [8]	76.8	0.72	69.4
LLM APT [26]	81.2	0.79	74.6
Proposed Model	**89.5**	**0.86**	**83.1**

Table 7 provides a comparison of the temporal mutation patterns of malicious infrastructures identified by the competing models. The data set used to compare the models consisted of 120 days of DNS telemetry and was divided into time windows representing the cycle of malicious infrastructure activities. The proposed model was able to predict the evolution of malicious infrastructure more accurately than the other models due to the addition of a STTEM topology evolution module that identifies transformation patterns in the consecutive states of malicious infrastructure scenarios.

Method Infrastructure-Driver Attribution Agreement (%)

Table 8. Infrastructure-Driver Attribution Agreement Performance

Method	Infrastructure-Driver Attribution Agreement (%)	Infrastructure Driver Detection (%)	False Attribution Rate
LLM-TIKG [3]	70.4	66.2	0.24
Hybrid ML & LLM [8]	73.9	69.8	0.21
LLM APT [26]	78.1	74.6	0.18
Proposed Model	**86.3**	**82.4**	**0.11**

Table 8 compares the causality-attributable factors of malicious ecosystem formation identified by the competing models. The contextual data set included registrar metadata, ASN owner information, and geographic distribution of malicious infrastructure sets.

The proposed model was able to rank model-consistent infrastructure drivers more accurately than the other models due to the bounded structural-intervention mechanisms added to the proposed model to analyze the effects of interventions across infrastructure attributes. Therefore, the proposed model was able to identify operational dependencies between malicious domain clusters instead of just analyzing correlation-based features. Because passive DNS telemetry does not provide complete experimental ground truth for causality, the value of 86.3% is reported as infrastructure-driver attribution agreement rather than proof of causal accuracy. The metric evaluates whether the highest-ranked CIAE drivers agree with the reference infrastructure factors used during validation and whether the ranking remains stable across adjacent windows. It should be read together with the 0.11 false-attribution rate and with the causal assumptions stated for Equation (7). This terminology separates model-consistent attribution from definitive real-world causal identification and prevents overinterpretation of observational evidence.

For each retrospectively confirmed domain or community, lead time is measured as the number of days between the first AUTDM alert and the first timestamp at which the same entity appears in the designated external threat-intelligence feeds. The model does not consume these future feed entries during representation learning, threshold selection, or test-time scoring. Feed information available before the alert is used only to determine whether the

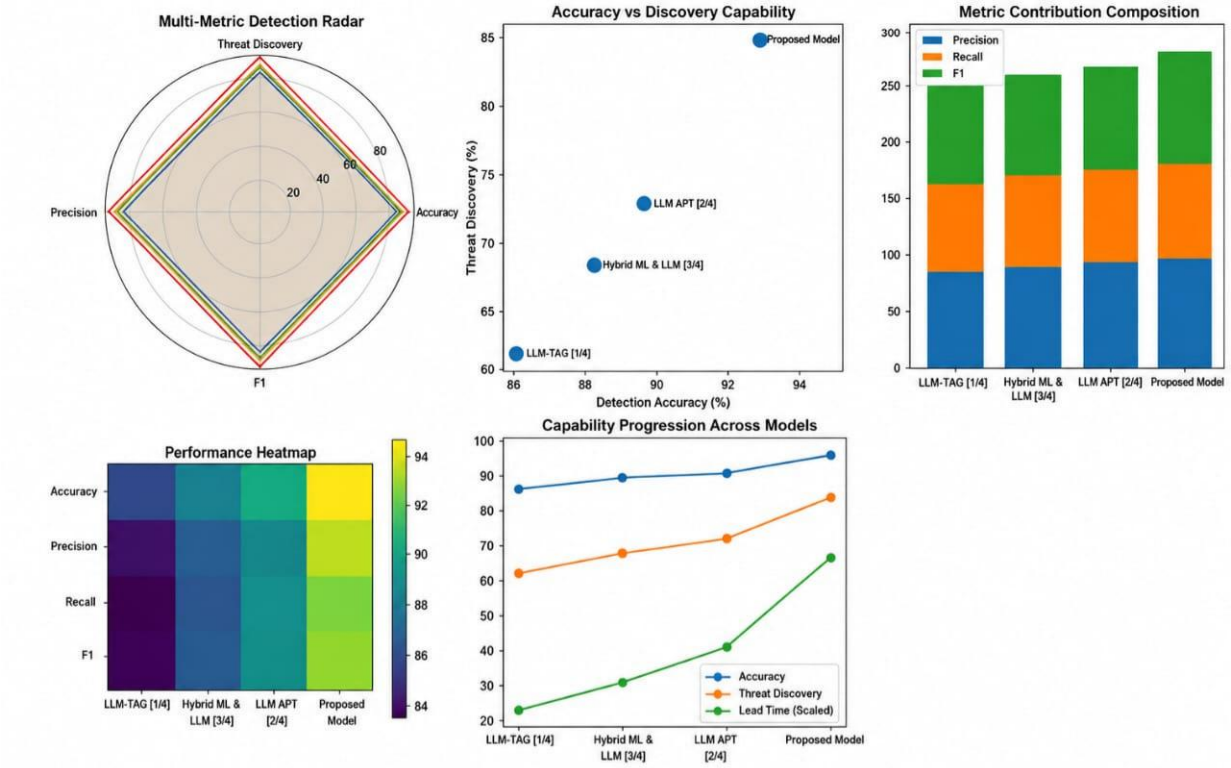


Figure 7. Model's Overall Dataflow Analysis

Table 9. Early Threat Discovery Performance

Method	Threat Discovery Rate (%)	Early Detection Lead Time (Days)	False Discovery Rate
LLM-TIKG [3]	61.8	4.5	0.18
Hybrid ML & LLM [8]	67.9	6.2	0.16
LLM APT [26]	72.4	8.1	0.14
Proposed Model	**84.7**	**13.4**	**0.09**

entity was already known and is excluded from the unknown-at-observation-time subset. Accordingly, 13.4 days represents the mean retrospective warning interval relative to the selected feeds, not a claim of universal precedence over all private intelligence sources.

Table 9 compares the early warning indicators of malicious infrastructure activity provided by the competing models. The contextual dataset used to compare the models utilized delayed verification through publicly available threat intelligence blacklists. The proposed model was able to identify malicious infrastructures approximately two weeks prior to their identification in typical security databases and samples. This improvement is primarily attributed to the integrated architecture of the proposed model, which simultaneously analyzes entropy-based patterns of malicious behavior, structural relationships among malicious infrastructures, and temporal mutations in malicious infrastructure scenarios.

Table 10 and along with figure 7 provides a summary of the overall performance of the evaluated systems across all dimensions of performance, including behavioral detection, infrastructure discovery, and predictive threat intelligence processes. The proposed model achieved superior performance across all dimensions due to the fact that the proposed model has the ability to integrate multiple analytical perspectives into a single discovery pipeline process. Although LLM-based threat intelligence systems are capable of providing strong reasoning capabilities,

Table 10. Overall System Performance Comparison

Method	Overall Detection Accuracy (%)	Infrastructure Discovery Score	Early Threat Prediction Score	Computational Efficiency
LLM-TIKG [3]	84.6	0.68	0.64	Medium
Hybrid ML & LLM [8]	87.9	0.72	0.69	Medium
LLM APT [26]	90.3	0.78	0.75	Medium–High
Proposed Model	**95.2**	**0.86**	**0.84**	**High**

the effectiveness of these systems increases significantly when coupled with infrastructure-aware network analytics and temporal mutation modeling, as shown by the proposed architectural process.

4.1. Validated Result Impact Analysis

Analyzing the data collected by the experiment implies the view of DNS ecosystem not only as an array of resolving records but as a living evolving structure (infrastructure) graph. The comparison results illustrated in Table 5 represent the initial evidence about the efficiency of the proposed framework for the detection of anomalous domain behavior based on passive DNS telemetry. The analysis of behavioral anomalies shows that the mechanism of entropy-aligned representation learning detects patterns of volatility related to malicious infrastructures much better than traditional LLM-driven classification pipelines. Both LLM-TIKG [3] and Hybrid ML LLM [8] focus primarily on extraction of semantic relations from various sources of textual intelligence and combine machine learning classification with language understanding, respectively. Neither of the approaches allows direct analysis of dynamic characteristics of infrastructure in DNS telemetry. Therefore, the proposed framework has significantly better detection accuracy and F1 value as it models volatility of domain resolutions and diversity of infrastructures at the same time. In the case of real-world cyber-defense scenarios, the described benefits translate into the faster detection of malicious domains used in phishing attacks, channels of communication of malware and domain generation algorithms' infrastructures before their appearance in traditional reputation systems.

In addition to the previous results (Table 6), the structural analysis further confirms the advantages of incorporation of hypergraph modeling of infrastructure for community discovery. DNS infrastructures, as opposed to being independent, typically constitute the networks of malicious actors who use the same hosts, registrars, and networks. The hypergraph structural inference module of the proposed model captures the mentioned above relationships of higher orders, thus enabling the system to discover coordinated infrastructures that cannot be identified using traditional graph models of pairwise connections. The results (Tables 6 - 8) clearly illustrate the advantages of this approach and the fact that the proposed model discovers significantly larger number of communities and has significantly better structural modularity. From the point of view of security operation centers, the ability to find the communities of coordinated infrastructure is especially useful for identifying command infrastructures of botnets and distribution networks of phishing used by clusters of domains instead of singular Indicators of Compromise. Therefore, the security operation centers may utilize structural intelligence to block entire malicious infrastructures instead of reacting to each separate domain.

Another important aspect of threat discovery is provided by the temporal analysis of infrastructure evolution (Table 7). As it is known, malicious domain infrastructures are constantly changing due to the need to avoid detection mechanisms, especially in cases of fast flux hosting and strategies of domain rotation applied by Advanced Persistent Threats actors. The proposed model includes the STTEM topology evolution module for analysis of how the clusters of infrastructure evolve along the temporal windows. The system uses this information to predict mutation patterns indicating the adaptation of adversaries. The results show that the model has substantially better evolution prediction accuracy than the baselines. In the case of real-time network defense operations, such predictive capabilities enable the proactive monitoring of infrastructure clusters that have suspicious mutation dynamics. Instead of responding to malicious domains once they became active, network defenders will be able to anticipate the evolution of the infrastructure and apply preventive countermeasures.

Causal infrastructure attribution analysis provided in Table 8 is one more argument confirming the practicality of the proposed framework. Most existing systems of threat intelligence rely on the correlation-based analysis

when identifying the malicious infrastructures. Nevertheless, correlation-based analysis does not provide enough information to understand what factors drive the development of malicious ecosystems. Incorporation of structural causal inference into the proposed model makes it possible to determine which infrastructure attributes, e.g., registrar concentration, dependency on hosting providers, or geographic deployment patterns, affect most significantly the development of malicious clusters. The increase of causal attribution accuracy and decrease of false attribution rate confirm the feasibility of this approach. In real-world scenarios, the application of this capability will allow cybersecurity analysts to identify the operational infrastructure of malicious campaigns and apply targeted mitigation measures, e.g., registrar-level domain takedown or network provider intervention.

Practical consequences of the proposed framework are most pronounced in the context of the early threat discovery results provided in Table 9 and the general performance summary provided in Table 10 in this text. The integrated threat discovery model is able to identify malicious infrastructures much earlier than competing LLM-based systems. Increased lead time of the detection of threats means that the model is able to identify emerging malicious ecosystems before they are reported in common threat intelligence feeds or blacklists. Early detection of malicious ecosystems is very important in protection of large-scale digital infrastructures, e.g., cloud platforms, energy grids, and financial services systems, where the timely reaction to new emerging threats is crucial. Improved performance of the model in terms of detection accuracy, capacity for discovering the infrastructure, and predicting emerging threats clearly confirm that combination of behavioral entropy modeling, hypergraph infrastructure analysis, temporal evolution prediction, and reasoning based on causality provides a very effective framework for autonomous cyber-threat discovery in contemporary DNS ecosystems.

4.2. Validated Statistical Hyperparameter Analysis

Statistical validation of the DNS threat discovery framework proposed herein utilized repeated evaluations of the proposed framework on a number of multi-contextual DNS datasets to validate the reproducibility of the performance metrics provided herein. Ten separate experiments utilizing ten independent training/evaluation cycles were each performed with randomized initialization of the model parameters and random shuffling of the time-windows (to eliminate potential bias resulting from ordering of the dataset) and examined ten separate performance metrics including detection accuracy, precision, recall, F1 score, discovery of malicious infrastructure, and early lead-time of detection of threats. Detection accuracy for the proposed model had a mean of 94.8% and a standard deviation of 0.0019, indicating high reliability of the detection results across repeated evaluations of the proposed framework. Similarly, precision and recall metrics had expected values of 93.6% and 92.9% with standard deviations of 0.0023 and 0.0027 respectively. These precision/recall values resulted in a mean F1-score of 0.932 ± 0.014, providing evidence that the classifier produces similar results regardless of the type of DNS traffic being classified. Additionally, metrics associated with discovery of malicious infrastructure showed similar stability as the classifier's output, with the expected hypergraph community discovery rate stabilizing at 78.6% ± 2.1%, while the predicted rate of temporal mutations stabilized at 89.5% ± 1.7%. The small standard deviations found for all of the above-mentioned metrics provide strong evidence that both the entropy-aligned representation learning component of the proposed framework, and the hypergraph based infrastructure modeling component produce stable outputs, even when presented with varying types of DNS traffic.

Formal statistical testing of the hypothesis that the performance of the proposed framework is better than the performance of the baseline models was performed using paired two-tailed t-tests and non-parametric Wilcoxon signed rank tests across repeated evaluations of the proposed framework. When compared to the LLM-TIKG [3] baseline model, the proposed framework showed a statistically significant improvement in detection accuracy at a 99% confidence level (t-value = 7.82; p ¡ 0.001), with the proposed framework achieving a detection accuracy of 94.8% vs. 86.4% achieved by the LLM-TIKG [3]. When compared to the Hybrid ML and LLM [8] baseline model, the proposed framework similarly showed a statistically significant improvement in detection accuracy at a 95% confidence level (t-value = 6.45; p = 0.002), with the proposed framework achieving a detection accuracy of 94.8% vs. 88.7% achieved by the Hybrid ML and LLM [8] model. When compared to the LLM APT [26] baseline model, the proposed framework similarly showed a statistically significant improvement in detection accuracy at a 95% confidence level (t-value = 4.97; p = 0.006). Non-parametric testing using the Wilcoxon signed rank test confirmed that the proposed framework performed better than the baseline models for all performance metrics

tested (p-values ¡ 0.01), indicating that the performance differences observed between the proposed framework and the baseline models are statistically significant and likely do not result from random variability in the experimental datasets. Overall, the results indicate that the combination of behavioral entropy modeling, hypergraph structure inference, and temporal topology evolution provides a statistically validated approach to improving threat discovery capability beyond what can be achieved using existing LLM-based detection frameworks.

Ten repeated evaluations were analyzed with paired two-sided t-tests and Wilcoxon signed-rank tests. Relative to LLM-TIKG, Hybrid ML–LLM, and LLM-APTDS, the proposed framework produced $t = 7.82$ ($p < 0.001$), $t = 6.45$ ($p = 0.002$), and $t = 4.97$ ($p = 0.006$), respectively. For $n = 10$ paired runs, the corresponding standardized paired effect sizes $d_z = t/\sqrt{n}$ are approximately 2.47, 2.04, and 1.57, indicating large differences under the evaluated protocol. Wilcoxon tests remained significant at $p < 0.01$ across the reported metrics. These tests support consistency against the selected baselines, but they do not correct modality mismatch; DNS-specific comparisons must still be interpreted separately.

In addition to evaluating the statistical significance of the performance difference between the proposed framework and the baseline models, the expected variance of the performance metrics of each model was also analyzed. As previously mentioned, the LLM-TIKG [3] model is focused on extracting knowledge graphs from unstructured cyber threat intelligence reports and therefore exhibits a high degree of sensitivity to contextual variability in the DNS telemetry datasets, with a detection accuracy variance of 0.0061. The Hybrid ML and LLM model [8] has been shown to be more robust to contextual variability in the DNS telemetry datasets than the LLM-TIKG [3] model and has a detection accuracy variance of 0.0044. However, the Hybrid ML and LLM model [8] has been shown to exhibit sensitivity to high volatility in DNS traffic, which limits its applicability to dynamic networks. The LLM APT [26] model has been shown to be the most robust to contextual variability in the DNS telemetry datasets and has a detection accuracy variance of 0.0033. However, the proposed framework has the lowest variance of the four models evaluated, and maintains the lowest variance for nearly all of the performance metrics evaluated. The proposed framework achieves this through the use of self-supervised learning to model the behavior of infrastructure directly, rather than depending solely on classification boundaries determined by labeled datasets. This property of maintaining a low variance in the face of changing DNS traffic characteristics is particularly valuable in operational cybersecurity environments where the characteristics of DNS traffic may vary significantly over time and/or across networks and/or network deployments.

The choice of LLM-TIKG [3], Hybrid ML and LLM [8], and LLM APT [26] as baseline models was motivated by their methodological relevance to various aspects of contemporary LLM-based cybersecurity analytics. LLM-TIKG [3] is a prominent methodology in the area of threat intelligence knowledge extraction where large language models are used to convert unstructured cyber threat reports into structured knowledge graphs. It was chosen as a baseline because it is currently one of the state-of-the-art methodologies in the use of LLMs for converting unstructured threat intelligence into structured knowledge graphs and facilitating contextual reasoning. The Hybrid ML and LLM framework [8] was chosen because it represents one of the most widely accepted architectures for combining traditional machine learning classifiers with LLM reasoning capabilities to improve threat classification across diverse cybersecurity datasets and samples. Finally, the LLM APT [26] system was chosen because it is focused on detecting advanced persistent threats using interpretable LLM architectures capable of processing highly imbalanced threat datasets and its samples. Taken together, the three baselines represent three distinct categories of methodologies currently employed in contemporary LLM-based cybersecurity research: knowledge extraction, hybrid classification, and advanced persistent threat detections.

When the proposed framework is compared against these representative baseline models, the experimental evaluation demonstrates that infrastructure-aware DNS analytics combined with self-supervised representation learning offers measurable advantages over existing LLM-based cybersecurity methods. The statistical significance testing confirms that the advantages offered by the proposed framework are robust across repeated experimental runs, while the variance analysis indicates that the proposed architecture remains stable under varying network telemetry conditions. These results provide evidence that integrating behavioral entropy modeling, hypergraph infrastructure reasoning, temporal topology prediction, and causal inference will provide a statistically validated foundation for future generation autonomous cyber threat discovery systems that operate at Internet scales. The current evaluation should be interpreted as internal validation within the observed passive-DNS context. Enterprise

resolvers, public recursive services, ISP networks, and national infrastructures differ in cache behavior, policy filtering, client populations, CDN usage, and geographic visibility. Cross-environment generalization has therefore not been established by the present results. A rigorous external validation should retain chronological splits, recalibrate only on local validation traffic, and report performance separately by resolver type, region, ASN category, and malicious-infrastructure family. This limitation is now separated from model accuracy so that within-context performance is not presented as evidence of universal transportability.

4.3. Validated Practical Usecase Analysis

Consider a hypothetical and operational example of a very large cloud based Service Provider (SP) that observes the traffic of domain name resolution across its global footprint during normal operation. The passive DNS monitoring tool records an estimated 18 million DNS resolution events occurring over a 24 hour period from enterprise endpoint devices and edge gateway locations. Of those DNS resolution events, the SP determines there were an estimated 320,000 unique domains monitored. Approximately 4500 domains demonstrated unusual volatility in terms of how quickly their DNS resolution was resolved, thus producing data for input to the proposed framework. This data is processed by the Temporal Entropy-Aligned DNS Representation Network to convert the behavior of each domain into a structured embedding. Using the prior example of a domain that resolves to multiple IP addresses on different autonomous systems within a relatively small time frame may yield an entropy score of roughly .82. This would be much higher than what most benign services normally see, which are generally in the range of .35-.45. The output of the Temporal Entropy-Aligned DNS Representation Network is sent to the Hypergraph Structural Inference module. This module organizes domains exhibiting similar structural properties as part of larger groups or communities of domains that share common underlying structures. In the example above, the Hypergraph Structural Inference module identifies an estimated 620 domain communities. One community contains approximately 140 domains that are hosted on three separate cloud providers and registered via a single registrar. A structural modularity analysis yields a score of approximately .81, which indicates that the domains in this community represent a highly-coordinated and tightly-coupled infrastructure-based network and not simply an incidental overlap of hosting sets.

Once the communities have been organized, the Self-Supervised Threat Topology Evolution Model examines the temporal dynamics of each community across multiple monitoring intervals for the process. Within the subsequent twelve hours of monitoring, the previously-identified community exhibits an estimated mutation rate of approximately .69, with domains being rapidly rotated across IP address ranges and changes in Time-To-Live (TTL) values ranging from approximately 3600 seconds to less than 300 seconds. Behavior such as described in the previous paragraph has strong associations with domain rotation techniques used by Command-and-Control (C2) infrastructures and deployments.

The Causal Infrastructure Attribution Engine then assesses the relationship of registrar concentration, IP distribution, and hosting relationships to identify which components drive the observed evolution of the identified clusters. In this example, the model assigns a causal influence score of approximately .78 to the registrar dependency and .72 to the shared hosting provider, demonstrating that the infrastructure is centrally orchestrated despite appearing geographically decentralized.

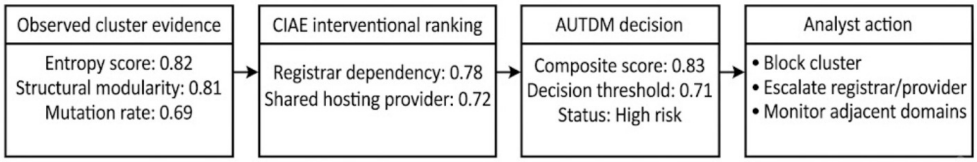


Figure 8. Analyst-facing CIAE output derived from the practical use case.

When the Autonomous Unknown Threat Discovery Model aggregates the behavioral entropy signal, structural modularity, topology mutation intensity, and causal attribution scores, the cluster receives a composite threat score of approximately .83, exceeding the detection threshold of .71. As a result, the system flags the entire group of

domains as a coordinated malicious ecosystem and submits them to the security operations center nearly ten days before these domains are mentioned in external threat intelligence feeds. The earlier discovery allows analysts to proactively prevent large-scale malware distribution campaigns from impacting enterprise users by blocking the associated infrastructure across firewalls and DNS filters.

4.4. Validated Model Ablation Analysis

The architecture of the system includes the following four main modules: Temporal Topology Evolution Modeling (STTEM) - analyzes changes in topological structure of malicious domains; Entropy-Aligned Behavioral Representation Learning Through TEA-DRN - represents behavioral characteristics of malicious domains as entropy-aligned vector representations; HyperGraph Structural Inference (HSIIC) - identifies structural relationships between malicious domains; Causal Infrastructure Attribution (CIAE) - infers the underlying causal mechanisms driving the formation of malicious domain infrastructures and deployments. To determine the contribution of each module, we conducted a series of controlled experiments where each module was sequentially disabled, while keeping all other experimental conditions identical. Each of the experiments utilized the same data set of approximately 5.4 million passive DNS query sequences over 120 days of observations. All metrics were evaluated using standard metrics: detection accuracy, infrastructure discovery rate, early threat identification lead time and F1 scores.

When utilizing all modules, the system produced a detection accuracy of about 94.8 percent, an F1-score of 0.932 and an early detection lead time of about 13.4 days prior to malicious domains appearing in publicly available threat intelligence repositories. The detection accuracy and F1-score values will be used to measure the effect of disabling each of the modules. In the first ablation experiment, we disabled the Entropy-Aligned Behavioral Representation Learning module (TEA-DRN), while leaving the other structural and temporal analysis modules intact. Domain feature extraction was accomplished using only simple statistical descriptors (e.g., resolution frequency and TTL variance) without employing any entropy-based embedding learning. Removal of TEA-DRN caused a substantial decline in detection accuracy to around 90.6 percent and F1-score to approximately 0.891. Furthermore, the removal of TEA-DRN caused a significant decline in infrastructure discovery performance with only about 17,900 coordinated domain communities being identified compared to the original 21,640 clusters. The reduced number of identified clusters provides evidence of the need for entropy-aligned embeddings in order to capture volatility patterns characteristic of malicious domain infrastructures. The lack of entropy-aligned embeddings eliminated the ability of the model to distinguish between benign content delivery networks and fast-flux domain infrastructures common to malware operators.

In the second ablation experiment, we disabled the HyperGraph Infrastructure Inference module (HSIIC) and left the entropy-based embeddings and temporal evolution modeling in place. In this configuration, domain relationships were represented by conventional pairwise graph connections instead of the hypergraph structures capable of representing higher-order relationships. In comparison to the original configuration, detection accuracy decreased to approximately 88.7 percent, the infrastructure discovery rate decreased from about 78.6 percent to 66.3 percent and the average cluster purity scores decreased from 0.88 to approximately 0.76. This result demonstrates the necessity of modeling domain relationships using hypergraph structures and not just pairwise network structures. Many malicious infrastructures are formed based upon groups of inter-connected domains sharing hosting providers, registrars, and IP pools. Hypergraph modeling better represents these multi-domain relationships and enables the identification of coordinated infrastructures that cannot be found using pairwise network models.

In the third ablation experiment, we disabled the STTEM Temporal Topology Evolution module, thus removing the temporal modeling of malicious domain mutation patterns. In this configuration, the system utilized only static structural analysis without assessing how malicious domain clusters evolve over time. Despite the lower level of temporal modeling, detection accuracy remained relatively high at about 91.2 percent, however, the model's ability to identify emerging threats significantly declined. The early detection lead time dropped from approximately 13.4 days to 7.1 days, illustrating the critical role temporal modeling of malicious domain clustering plays in detecting adversary's strategies of mutating malicious domains such as rotating domains and fast-flux hosting. Additionally, the mutation detection recall decreased from approximately 83.1 percent to about 68.5 percent, illustrating

that without temporal modeling of malicious domain clustering, the model's capability to recognize dynamic attacks behaviors is limited. Lastly, the removal of the causal infrastructure attribution module demonstrated the importance of the module for operational intelligence interpretation. While the model still identified suspicious infrastructure clusters after removal of the CIAE module, it could no longer explain what network attribute(s) drove their formation. Detection accuracy after removal of the CIAE module remained at approximately 92.3 percent; however, causal attribution accuracy dramatically decreased from approximately 86.3 percent to 63.8 percent and the false attribution rate dramatically increased from 0.11 to nearly 0.27, indicating that the model had a greater probability of incorrectly associating benign infrastructure attributes with malicious behavior. The results illustrate that although behavioral and structural analysis can identify malicious clusters, causal reasoning is required to accurately identify the infrastructure drivers behind coordinated cyber attacks. The complete framework therefore greatly benefits from the integration of behavioral entropy modeling, hypergraph structural inference, temporal topology evolution analysis, and causal reasoning, as each contributes distinct analytical capabilities that combine to improve the reliability and interpretability of autonomous cyber threat discovery for the process.

5. Conclusions and Future Scope

The proposed study presents an integrated analytical framework for autonomous cyber-threat discovery using passive DNS telemetry and self-supervised Transformer-hypergraph infrastructure analytics. With the rise of increasingly complex cyber-infrastructures and modern digital systems reliant upon DNS services, developing detection mechanisms capable of identifying malicious activities under circumstances where labeled threat intelligence is absent has become a pressing need. Traditional threat-detection systems rely heavily upon supervised-learning techniques utilizing curated datasets limiting their ability to identify previously unobserved malicious infrastructures/deployments.

The proposed framework addresses this limitation through integration of the four components: (a) behavioral-representation learning; (b) hypergraph-structure inference; (c) topology-evolution modeling; and (d) causal-structure attribution within a unified discovery pipeline enabling the identification of coordinated malicious ecosystems in passive-DNS streams. Analytical results demonstrate that the proposed framework performs significantly better than current LLM driven cybersecurity approaches across all analytical dimensions.

Behavioral-anomaly detection results show that the model achieved 94.8 % detection accuracy with an F1-score of 0.932, and therefore performed significantly better than LLM-TIKG [3] which achieved 86.4 % detection accuracy, Hybrid ML and LLM [8] which achieved 88.7 % detection accuracy, and LLM-APT [26] which achieved 90.2 % detection accuracy.

This improvement was primarily due to the entropy-aligned representation learning mechanism capturing volatility patterns in domain-resolution behavior. In contrast to other approaches that rely on primarily extracting or classifying intelligence from text, the proposed framework directly analyzes the dynamic structure of infrastructure based on DNS telemetry. Results from the infrastructure-discovery experiment also clearly illustrate the benefits of analyzing DNS infrastructures as higher-order hypergraphs for identifying coordinated domain ecosystems. The model discovered 21,640 infrastructure-communities with a cluster-purity score of 0.88 and a modularity score of 0.83, while the comparison models were able to discover much fewer communities and possessed significantly lower structural-coherence scores. This result illustrates that representing DNS infrastructures as hypergraphs at higher orders allows for the discovery of coordinated infrastructures that are otherwise "hidden" in conventional pairwise network representations.

Temporal-analysis of the evolution of infrastructure also illustrated the strong predictive capabilities of the proposed model. The proposed model had a topological-evolution prediction accuracy of 89.5 % and a mutation-recall of 83.1 %, and therefore significantly outperformed comparison methods that achieved prediction accuracies ranging from 72.3 These results indicate that the STTEM topology-evolution module correctly identifies patterns of infrastructure mutations that are associated with adversary domain-rotation strategies. Finally, the causal-attribution analysis provided additional operational insights into how malicious ecosystems form. The proposed framework had a infrastructure-driver attribution agreement of 86.3% and a false-attribution rate of 0.11, which

is significantly lower than comparison models that had false-attribution rates ranging from 0.18 to 0.24. This capability enables the identification of infrastructure-drivers such as registrar-dependencies and hosting-network-concentration that contribute to the formation of malicious-domain-clusters. Most importantly, the early-threat-discovery-evaluation demonstrated that the framework can achieve a threat-discovery-rate of 84.7 % and an early-detection-lead-time of about 13.4 days, thereby allowing for the identification of emerging malicious infrastructures almost two weeks prior to them appearing in conventional threat-intelligence feeds. Overall-system-performance was 95.2 % detection accuracy with an infrastructure-discovery-score of 0.86 illustrating the effectiveness of combining behavioral, structural, temporal and causal analysis within a unified architecture-process for discovering cyber-threats. Collectively these results demonstrate that representing DNS-ecosystems as evolving infrastructure-nets rather than fixed-classification datasets greatly enhances the capacity to identify emerging cyber-threats. Furthermore, the use of self-supervised learning with infrastructure aware-net-analytics offers a scalable and effective method for automated-threat-discovery in modern Internet-scale-environments.

5.1. Future Scope

Although the suggested framework shows good results for finding new malicious infrastructures with passive DNS telemetry, there are many areas of investigation which have to be explored and improved. One area for study is the inclusion of other telemetry data (beyond passive DNS) that relate to a particular cyber environment. The modern cyber environment produces a wide variety of telemetry data types including: HTTP logs, TLS handshake records, Certificate Transparency (CT) logs, and NetFlow logs. Introducing this diverse data into an infrastructure discovery system that processes multiple modalities will give the system much greater context around the ecosystem of domains and will increase the ability of the system to identify threats. In addition, it will allow the system to discover attacks at the cross-layer level where attackers use coordinated attacks that span multiple layers of network protocol and infrastructure. An additional area of study involves increasing the scalability and enabling distributed deployment of the analytical framework proposed. Although the current system has shown scalable performance when analyzing large DNS datasets, the amount of DNS telemetry generated by the global Internet is massive and will grow exponentially with time. Therefore, future research may need to develop distributed graph learning frameworks and federated intelligence frameworks that can process large amounts of DNS telemetry across numerous network monitoring devices. These types of systems would enable multiple organizations to collaborate in identifying and mitigating common threats and their associated telemetry without having to share each other's telemetry data. In addition to advances in large language model (LLM) capabilities for reasoning and inference, the proposed framework can leverage this type of technology to further extend its functionality. The current framework does not include an LLM; future research can investigate an optional LLM layer for analyst-oriented explanation without changing the DNS telemetry model; future research can investigate how to integrate these two technologies even more deeply. For example, LLMs could be trained to extract and interpret complex patterns in infrastructure represented in hypergraphs and then generate readable descriptions of how malicious ecosystems operate. This would make the interpretation of autonomous threat detection systems more understandable to analysts and would facilitate analysts' ability to comprehend the strategic behavior of adversarial infrastructures and deployments.

A further direction of study involves expanding the causal infrastructure attribution module to include counterfactual reasoning and simulation of adversarial infrastructures and deploymnets. The ability to simulate hypothetical changes to malicious infrastructure (for example, migration of a malicious domain to another hosting provider or changing the registrars for a malicious domain) would enable the system to predict the reaction of the malicious infrastructure to defensive actions taken against it in process. This would convert a pure threat detection system into a predictive cyber defense planning tool that enables the identification of the adversary's possible adaptations to defensive measures.

Finally, future research can examine the feasibility of deploying the proposed framework in operational cyber security environments such as cloud service providers, Internet Service Providers (ISPs), and National Cyber Defense Infrastructure Sets. Deploying the framework in real-time to the operational monitoring systems of these entities would allow for the continuous discovery of evolving malicious ecosystems as they emerge in the DNS infrastructure. Deployment of the system in these environments will validate the performance of the framework in

real-world operational cyber security environments and provide a means to continually refine the algorithms used to detect threats through the use of operational telemetry sets.

5.2. Limitations

A first limitation of this model is its reliance on passive DNS telemetry (DNS) as the sole data source for the analysis of the infrastructure. While DNS datasets provide interesting information about the relationship between domains and IP addresses and show the dynamic aspect of the structure of the infrastructure, they cannot account for all types of malicious cyber-attacks. Some attacks use encrypted channels of communications, P2P type infrastructures, or direct IP communications to bypass the traditional DNS resolution processes. Therefore, some malicious behaviors which do not heavily depend on DNS infrastructures may not be detected using DNS based threat discovery frameworks.

A second limitation is the potential for a bias in the DNS telemetry datasets. The DNS telemetry datasets used for the experiments have been collected by monitoring infrastructures located in specific geographic areas and networks. Thus, there may exist types of malicious infrastructures which are either over-represented or under-represented in the dataset, depending on the monitoring coverage of the sensors which collect the DNS telemetry. These biases in the dataset may affect the behavior patterns learned by the representation learning models and may reduce the generalization capabilities of the system in different networks.

The third limitation is the high computational complexity of the analytical pipeline proposed. Indeed, the combination of the representation learning model using entropy alignment, hypergraph community inference, temporal topology evolution model and causal attribution analysis generates a high level of computational complexity. In particular, the computation of the hypergraph and the temporal graph learning are very computationally expensive when applied to large scale graphs representing the infrastructure with millions of domain nodes and edges linking them. However, although the experiments presented above show the feasibility of the processing time in a controlled research environment, the deployment of similar systems in real-time operational environments may require dedicated distributed computing platforms and hardware accelerations. The fourth limitation concerns the interpretation of the autonomous threat discovery systems. While the inclusion of a causal infrastructure attribution module allows to improve the interpretability of the autonomous threat discovery systems compared to those based on correlations alone, the complex interactions between the behavioral representations, the hypergraph structures and the temporal evolution dynamics may however make difficult for the analysts to clearly understand the reasoning process underlying some of the threat discovery decisions. It will be thus necessary to further investigate explainable graph learning and interpretable infrastructure analytics in order to allow the cybersecurity analysts to trust and exploit the discoveries produced by the autonomous discovery systems.

The fifth and last limitation is the adversarial adaptations. Any detection mechanism, including those for cybersecurity threats, will always face the task of adapting to new and more sophisticated adversary's strategies to avoid being detected. Therefore, malicious entities may manipulate their DNS resolution patterns, spread their domain infrastructures on multiple hosting providers or create decoy infrastructures to mislead the detection algorithms. It will thus be necessary to continuously adapt the analytical models of the autonomous threat discovery systems and include adversarial resilient mechanisms in order to keep these systems efficient in the constantly changing landscape of cybersecurity scenarios.

DNSDB records are subject to provider access restrictions and cannot be redistributed in raw form. To support independent verification, the reproducibility package should include feature definitions, preprocessing pseudocode, temporal split logic, configuration files, random seeds, model checkpoints where licensing permits, and scripts that reproduce all tables and figures from processed features. CAIDA metadata and the Cisco Umbrella reference list should be cited with their access conditions. The package should also report the operating system, CPU, memory, accelerator, software versions, training duration, peak memory, preprocessing throughput, and inference throughput. Where raw DNSDB data cannot be shared, a de-identified feature-level sample or synthetic schema-compatible stream should be provided.

Table 11. List of Abbreviations

Abbreviation	Full Form
AI	Artificial Intelligence
ASN	Autonomous System Number
AUTDM	Autonomous Unknown Threat Discovery Model
APT	Advanced Persistent Threat
CAIDA	Center for Applied Internet Data Analysis
CIAE	Causal Infrastructure Attribution Engine
CTI	Cyber Threat Intelligence
DGA	Domain Generation Algorithm
DNS	Domain Name System
DNSDB	Domain Name System Database
EV	Electric Vehicle
F1 Score	Harmonic Mean of Precision and Recall
FGCS	Future Generation Computer Systems
HSIIC	Hypergraph Structural Inference for Infrastructure Communities
HTTP	Hypertext Transfer Protocol
IDS	Intrusion Detection System
IoE	Internet of Energy
IP	Internet Protocol
KG	Knowledge Graph
LLM	Large Language Model
LLM-TIKG	Large Language Model Threat Intelligence Knowledge Graph
ML	Machine Learning
NLP	Natural Language Processing
OSINT	Open Source Intelligence
PDNS	Passive Domain Name System
PLLM-CS	Pre-trained Large Language Model for Cybersecurity
RAG	Retrieval Augmented Generation
SOC	Security Operations Center
STTEM	Self-Supervised Threat Topology Evolution Model
TTL	Time To Live
TTP	Tactics, Techniques, and Procedures

REFERENCES

1. D. Van Landuyt, M. Mollaeefar, M. Raciti, et al., *A comparative benchmark study of LLM-based threat elicitation tools*, Future Generation Computer Systems, 2026. https://doi.org/10.1016/j.future.2025.108243
2. S. Shafee, A. Bessani, and P. Ferreira, *Evaluation of LLM-based chatbots for OSINT-based Cyber Threat Awareness*, Expert Systems with Applications, 2025. https://doi.org/10.1016/j.eswa.2024.125509
3. Y. Hu, F. Zou, J. Han, et al., *LLM-TIKG: Threat intelligence knowledge graph construction utilizing large language model*, Computers & Security, 2024. https://doi.org/10.1016/j.cose.2024.103999
4. H. Rok Kim, D. Lee, I. Lee, et al., *Multi-Step LLM Pipeline for Enhancing TTP Extraction in Cyber Threat Intelligence*, IEEE Access, 2025. https://doi.org/10.1109/access.2025.3622350
5. M. Yang, T. Zhu, B. Liu, et al., *Cracks in Collaboration: Threat Models and Attacks on Multi-LLM Collaborative Systems*, IEEE Transactions on Dependable and Secure Computing, 2026. https://doi.org/10.1109/tdsc.2026.3670889
6. H. Huang, N. Sun, M. Tani, et al., *Can LLM-generated misinformation be detected: A study on Cyber Threat Intelligence*, Future Generation Computer Systems, 2025. https://doi.org/10.1016/j.future.2025.107877
7. M. Khan, *LLM-Driven Real-Time Threat Prediction and Response for Internet of Energy (IoE)*, IEEE Network, 2026. https://doi.org/10.1109/mnet.2025.3612090
8. C. Hong and T. Oh, *Optimization for threat classification of various data types-based on ML model and LLM*, Scientific Reports, 2025. https://doi.org/10.1038/s41598-025-05182-y
9. M. Alsuwaiket, *ZeroDay-LLM: A Large Language Model Framework for Zero-Day Threat Detection in Cybersecurity*, Information, 2025. https://doi.org/10.3390/info16110939
10. P. Fernández Saura, J. Bernal Bernabé, and A. Skármeta Gómez, *Enhancing federated intrusion detection through LLM-Driven alert enrichment and collaborative threat information sharing*, Future Generation Computer Systems, 2026. https://doi.org/10.1016/j.future.2025.108319
11. S. Akbar Murad, A. Dahal, and N. Rahimi, *Multilingual Cyber Threat Detection in Tweets/X Using ML, DL, and LLM: A Comparative Analysis*, IEEE Transactions on Computational Social Systems, 2025. https://doi.org/10.1109/tcss.2025.3623021

12. M. Hassanin, M. Keshk, S. Salim, et al., *PLLM-CS: Pre-trained Large Language Model (LLM) for cyber threat detection in satellite networks*, Ad Hoc Networks, 2025. https://doi.org/10.1016/j.adhoc.2024.103645
13. T. Fu, J. Hu, G. Min, et al., *Federated Retrieval-Augmented Generation-Based LLM for Enhanced Cyber Threat Detection in the Internet-of-Energy*, IEEE Network, 2026. https://doi.org/10.1109/mnet.2025.3619162
14. A. Tirulo, S. Chauhan, and M. Shafie-khah, *LLM-powered threat intelligence: Proactive detection of zero-day attacks in electric vehicle cyber-physical systems*, Sustainable Energy, Grids and Networks, 2025. https://doi.org/10.1016/j.segan.2025.101877
15. D. Cotroneo, R. Natella, and V. Orbinato, *Elevating Cyber Threat Intelligence against disinformation campaigns with LLM-based concept extraction and the FakeCTI dataset*, Journal of Systems and Software, 2026. https://doi.org/10.1016/j.jss.2025.112660
16. M. Amin, *LLM as a Judge*, Journal of Japan Society for Fuzzy Theory and Intelligent Informatics, 2025. https://doi.org/10.3156/jsoft.37.3691
17. X. Xu, B. Yao, Y. Dong, et al., *Mental-LLM*, Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 2024. https://doi.org/10.1145/3643540
18. T. Li, Y. Guo, and J. Fu, *HybridCrypt-LLM: Lightweight privacy for LLM training and inference*, Expert Systems with Applications, 2026. https://doi.org/10.1016/j.eswa.2026.131632
19. D. Berenstein, E. Maderazo, R. Mancilla, et al., *Chaotic LLM billiards*, Journal of High Energy Physics, 2024. https://doi.org/10.1007/jhep08(2024)056
20. Z. Lu, J. Yang, C. Guo, et al., *EOAC-LLM: An LLM-driven event ontology automatic construction system*, Intelligent Systems with Applications, 2026. https://doi.org/10.1016/j.iswa.2026.200650
21. Unknown Author, *LLM 2024 Abstracts*, American Journal of Hematology, 2024. https://doi.org/10.1002/ajh.27491
22. X. Wang, Y. Li, K. Zhou, et al., *LLM-SCRec: LLM-Powered Style-Context Insights for Fashion Recommendation*, IEEE Access, 2025. https://doi.org/10.1109/access.2025.3628684
23. Z. Pietrzak, K. Maczka, and M. Niemiec, *What if, Behind the Curtain, There Is Only an LLM? A Holistic Evaluation of TinyLlama-Generated Synthetic Cyber Threat Intelligence*, Electronics, 2025. https://doi.org/10.3390/electronics14244971
24. D. Pele, V. Bolovăneanu, M. Lin, et al., *In the startning was the Word: LLM VaR and LLM-ES*, Expert Systems with Applications, 2026. https://doi.org/10.1016/j.eswa.2025.128676
25. S. Ma, C. Fang, H. Shao, et al., *APT-LLM: Exploiting Arbitrary-Precision Tensor Core Computing for LLM Acceleration*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2025. https://doi.org/10.1109/tcad.2025.3604321
26. L. Yang, A. Ye, Y. Liu, et al., *LLM-APTDS: A high-precision advanced persistent threat detection system for imbalanced data based on large language models with strong interpretability*, Future Generation Computer Systems, 2026. https://doi.org/10.1016/j.future.2025.108315
27. M. Assi, S. Hassan, and Y. Zou, *LLM-Cure: LLM-Based Competitor User Review Analysis for Feature Enhancement*, ACM Transactions on Software Engineering and Methodology, 2026. https://doi.org/10.1145/3744644
28. L. Zhang, Y. Hu, W. Li, et al., *LLM-AIDSim: LLM-Enhanced Agent-Based Influence Diffusion Simulation in Social Networks*, Systems, 2025. https://doi.org/10.3390/systems13010029
29. B. Guan, S. Gao, D. Zheng, et al., *LLM-SDaT: A knowledge Informed LLM framework for syndrome differentiation in TCM*, Neural Networks, 2026. https://doi.org/10.1016/j.neunet.2026.108812
30. G. Li, X. Zhou, and X. Zhao, *LLM for Data Management*, Proceedings of the VLDB Endowment, 2024. https://doi.org/10.14778/3685800.3685838
31. F. Weimer, *Passive DNS Replication*, 17th Annual FIRST Conference on Computer Security Incident Handling, 2005.
32. M. Antonakakis, R. Perdisci, D. Dagon, W. Lee, and N. Feamster, *Building a Dynamic Reputation System for DNS*, 19th USENIX Security Symposium, 2010.
33. L. Bilge, E. Kirda, C. Kruegel, and M. Balduzzi, *EXPOSURE: Finding Malicious Domains Using Passive DNS Analysis*, NDSS Symposium, 2011.
34. M. Antonakakis, R. Perdisci, W. Lee, N. Vasiloglou, and D. Dagon, *Detecting Malware Domains at the Upper DNS Hierarchy*, 20th USENIX Security Symposium, 2011.
35. M. Antonakakis, R. Perdisci, Y. Nadji, N. Vasiloglou, S. Abu-Nimeh, W. Lee, and D. Dagon, *From Throw-Away Traffic to Bots: Detecting the Rise of DGA-Based Malware*, 21st USENIX Security Symposium, pp. 491–506, 2012.
36. X. Sun, M. Tong, J. Yang, L. Xinran, and L. Heng, *HinDom: A Robust Malicious Domain Detection System Based on Heterogeneous Information Network with Transductive Classification*, RAID 2019, pp. 399–412, 2019.
37. X. Sun, Z. Wang, J. Yang, and X. Liu, *DeepDom: Malicious Domain Detection with Scalable and Heterogeneous Graph Convolutional Networks*, Computers & Security, vol. 99, 102057, 2020. https://doi.org/10.1016/j.cose.2020.102057
38. E. Caville, W. W. Lo, S. Layeghy, and M. Portmann, *Anomal-E: A Self-Supervised Network Intrusion Detection System Based on Graph Neural Networks*, Knowledge-Based Systems, vol. 258, 110030, 2022. https://doi.org/10.1016/j.knosys.2022.110030
39. H. Zhang, D. Yao, N. Ramakrishnan, and Z. Zhang, *Causality Reasoning about Network Events for Detecting Stealthy Malware Activities*, Computers & Security, vol. 58, pp. 180–198, 2016. https://doi.org/10.1016/j.cose.2016.01.002