Semantic-Aware Anomaly Detection for Satellite-IoT Networks: A Lightweight Transformer-Based Approach

Junbeom Park
Dept. of Computer Engineering
Korea Aerospace University
Goyang, South Korea
jbpark@kau.kr

Zizung Yoon
Dept. of Smart Drone Engineering
Korea Aerospace University
Goyang, South Korea
z.yoon@kau.ac.kr

Taehoon Eom
AI Semiconductor Team
Artificial Intelligence Industry Cluster Agency
Gwangju, South Korea
eomth86@aicluster.or.kr

Jongsou Park
Dept. of Computer Engineering
Korea Aerospace University
Goyang, South Korea
jspark@kau.ac.kr

***Abstract*—Satellite-IoT networks are being increasingly deployed in mission-critical domains such as disaster response, military communications, maritime surveillance, and remote sensing. However, their heterogeneous architectures and resource-constrained nodes expose them to packet-level threats exploiting semantic dependencies across structured packet fields. Traditional intrusion detection systems (IDSs) often fail to capture such dependencies, particularly when packet fields are missing or incomplete. To address this challenge, we propose a lightweight anomaly detection approach based on DistilBERT—a compact Transformer-based language model fine-tuned to classify sentence-based representations of structured Satellite-IoT packets. The proposed sentence-based representation preserves inter-field dependencies and contextual semantics while enabling efficient processing in resource-constrained Satellite-IoT environments. A scenario-driven dataset was constructed to support this approach, incorporating 15 protocol- and security-aware fields derived from realistic communication flows. It includes three attack categories (injection, replay, and privilege abuse) and one Normal class, simulating diverse traffic conditions observed in operational Satellite-IoT environments. Experimental evaluations confirm that the proposed model accurately detects semantic anomalies under both complete and missing-field conditions, achieving 99.0% accuracy and a 98.9% F1-score. These results confirm the feasibility and practicality of applying a lightweight large language model (LLM) for semantic packet analysis in space communication systems and contribute to interpretable, context-aware intrusion detection in next-generation Satellite-IoT architectures.**

***Index Terms*—Satellite-IoT Networks, cybersecurity, anomaly detection, lightweight transformers, packet classification**

I. Introduction

Satellite-IoT networks support mission-critical applications such as maritime surveillance, disaster response, climate monitoring, and defense communications by linking space, ground, and user segments [1]. These systems connect satellites with terrestrial control infrastructure, including gateways and firewalls, and with user-side platforms such as Internet-of-Things (IoT) sensors, unmanned aerial vehicles (UAVs), and global navigation satellite system (GNSS) receivers operating across diverse operational and regulatory contexts [2]. Moreover, IoT devices positioned at the edge of Satellite-IoT networks frequently operate with outdated firmware and insufficient authentication mechanisms, exposing them as vulnerable endpoints and expanding the attack surface for threat actors [3], [4]. Compared with conventional terrestrial systems, Satellite-IoT environments must contend with bandwidth limitations, intermittent connectivity, and constrained device capabilities; these factors significantly complicate end-to-end security deployment [5]. These architectural and operational constraints hinder consistent security enforcement and fragment trust boundaries across ground, space, and user segments, thereby increasing the risk that vulnerabilities in one layer may propagate through others due to protocol heterogeneity. One illustrative case is the 2022 Viasat KA-SAT breach, in which attackers exploited a ground-based management network to distribute malicious firmware, ultimately disrupting satellite modem functionality across Europe [6]. This incident demonstrated how a compromise at the ground level can reverberate through the space-ground communication infrastructure. Such risks remain prevalent throughout the Satellite-IoT ecosystem, particularly when firmware, protocol configurations, or identity credentials are insufficiently protected [7].

Beyond infrastructure-level exploits, Satellite-IoT networks are increasingly exposed to semantic-level threats that exploit contextual inconsistencies across packet fields [8]. These anomalies—such as orbit-region mismatches, unauthorized control attempts, and replayed telemetry—often appear syntactically well-formed yet semantically inconsistent with operational logic, thereby evading conventional parsing-based checks [9], [10]. These threats reveal a core limitation of conventional intrusion detection systems (IDSs): their inability to model semantic relationships among packet fields. Although deep learning architectures such as recurrent neural networks (RNNs) and Transformers offer stronger representational capabilities [9], [11], they typically introduce significant computational overhead and struggle to interpret semantic relationships when input fields are incomplete or noisy [12]. As a result, persistent challenges remain for prevailing detection frameworks, especially within resource-constrained Satellite-IoT infrastructures. To overcome these limitations in capturing semantic inconsistencies, we propose a semantic-aware detection approach that leverages structured packet formats and Transformer-based reasoning. The method utilizes DistilBERT [13], chosen for its efficiency and compatibility with resource-constrained satellite platforms. Each structured packet is encoded as a natural-language-style sentence that reflects cross-field semantics, allowing the model to infer dependencies without relying on protocol-specific encoders. We construct a domain-specific dataset informed by realistic traffic patterns and attack behaviors, incorporating Satellite-IoT packet specifications (e.g., CubeSat Space Protocol (CSP) [14], MIOTY [15], and Consultative Committee for Space Data Systems (CCSDS) [16]), field schemas from public security datasets (e.g., TON-IoT [5]), and operational constraints derived from protocol rules and communication flows—such as orbit-region bindings, time-to-live (TTL) ranges, and port configurations. The dataset also includes adversarial cases involving contextual violations, including field tampering, unauthorized command injection, and payload-role inconsistencies.

This work offers three main contributions. First, we propose a semantic encoding scheme for structured packet data, enabling Transformer-based inference without additional preprocessing pipelines. Second, we demonstrate that a large language model (LLM) can effectively detect multi-field semantic anomalies, even under partial or degraded packet conditions. Third, we introduce a reproducible data generation methodology that produces a scenario-driven, labeled dataset tailored for semantic anomaly detection in Satellite-IoT environments. Unlike conventional IDSs based on rule matching, machine learning, or deep learning techniques, our method leverages language modeling to capture contextual inconsistencies across multiple structured packet fields in an integrated approach suitable for Satellite-IoT scenarios. The remainder of this paper is organized as follows. Section II reviews related work, Section III outlines the proposed detection approach, Section IV presents experimental results, and Section V concludes with a summary and future directions.

II. Related Work

Prior studies on intrusion detection for Satellite-IoT systems have explored protocol-level behavior analysis across heterogeneous segments in response to increasingly complex packet-level threats. This section reviews two key areas of related work: intrusion detection techniques for Satellite-IoT network security and the application of LLMs for semantic anomaly detection in cyber-physical communication environments.

A. Security of Satellite-IoT Networks and Intrusion Detection

Research on the security of Satellite-IoT networks has examined various intrusion detection mechanisms designed for heterogeneous protocols and resource-constrained nodes spanning integrated terrestrial and space-based systems. Alsaedi *et al.* introduced the TON_IoT dataset as a telemetry corpus tailored for IoT and Industrial IoT (IIoT) environments, offering structured features to support intrusion detection research [5]. While this dataset provides a valuable starting point, its applicability to satellite communication remains limited due to the absence of orbital context and satellite-specific semantics. Wang *et al.* applied co-attention and entropy-based mechanisms for anomaly detection in satellite telemetry streams, demonstrating strong performance on structured sequence inputs [17]. However, these methods may be less effective under dynamic protocol-level changes or degraded packet structures.

Lightweight IDSs for CubeSats have also gained traction as a complementary direction, given their strict constraints on memory, processing, and autonomy. Driouch *et al.* conducted a survey on CubeSat IDS frameworks that prioritized autonomy and low computational overhead [18], and later proposed a distributed IDS based on deep learning for classifying CSP-based packets [8]. These approaches have contributed significantly to the CubeSat security domain; nevertheless, the broader challenge of addressing semantic diversity across layers and message formats remains unresolved.

Complementary research on deep learning for IoT security, such as the work by Aldhaheri *et al.*, has proposed hybrid detection frameworks that incorporate multiple modalities [19]. While prior methods have shown reasonable performance in general IoT and satellite contexts, many rely on rigid data formats and struggle with semantic interpretation—particularly in the presence of ambiguous, degraded, or partially missing field data. Consequently, these limitations highlight the need for IDSs that extend beyond packet-level features. Detecting adversarial, semantically inconsistent, or partially missing field data requires models—such as LLM-based architectures—that can capture dependencies across structured packet fields.

B. LLM-Based Anomaly Detection

LLMs have emerged as a promising paradigm for network security due to their ability to capture semantic relationships across diverse protocol fields and traffic patterns [20]. Recent studies have explored integrating LLMs into anomaly detection pipelines targeting both IoT and satellite communication networks. Hassanin *et al.* [9] introduced PLLM-CS, a pretrained Transformer-based model designed for threat detection in satellite networks, integrating protocol-aware features with hierarchical threat modeling concepts. While the approach effectively captured satellite-specific attack vectors, it relied on synthetic and publicly available datasets, limiting its representativeness for real-world packet flows. Ferrag *et al.* [21] proposed a privacy-preserving LLM-based intrusion detection framework for IoT and IIoT devices. Their model, based on a lightweight BERT architecture, demonstrated high accuracy and scalability while preserving on-device privacy; however, the evaluation was confined to generic IoT settings without addressing protocol heterogeneity or orbital communication constraints. Worae *et al.* [22] proposed a unified framework for IoT management and traffic anomaly detection that combines contextual modeling with recent advances in LLMs, emphasizing explainability and cross-layer integration. However, real-time operational validation remains limited.

Collectively, these studies demonstrate the growing applicability of LLMs in cyber threat detection across both structured and unstructured telemetry. However, prior approaches often rely on rule-based or machine learning (ML) models that fail to capture semantic dependencies across packet fields, or they apply deep models with prohibitive inference costs for spaceborne systems. Some LLM-based methods exist, but they primarily target general-purpose datasets and lack adaptation to structured, domain-specific telemetry in Satellite-IoT networks. To address these limitations, we propose a compact LLM-based method tailored for semantic anomaly detection in Satellite-IoT environments. The approach leverages sentence-based representations of structured packets to support robust detection under incomplete, degraded, or adversarial conditions.

III. Proposed Methodology

This section presents a semantic anomaly detection approach for Satellite-IoT systems, which integrates domain-driven packet simulation, sentence-based input construction, and lightweight LLM-based classification to identify field-level anomalies under constrained communication environments. Building upon prior work in hierarchical segmentation and threat modeling [2], the proposed approach processes structured Satellite-IoT packets composed of 15 fields aligned with real-world protocols, including CSP, MIOTY, CCSDS, and TON_IoT.

The 15 fields used in packet construction are summarized in Table I. These include temporal attributes (`timestamp`); node-level identifiers (`src`, `dst`, `src_region`, `dst_region`); link-level parameters (`priority`, `src_port`, `dst_port`); orbital context (`orbit_class`); and message semantics (`msg_type`, `payload_type`, `payload`, `label`, `ttl`, and `flags`). Each field is designed to reflect protocol-compliant structures and operational semantics across space, ground, and user segments, ensuring compatibility with CSP, MIOTY, CCSDS, and TON_IoT specifications.

Collectively, these fields encode both syntactic and semantic dimensions of packet metadata—bridging low-level protocol syntax with higher-level contextual semantics—and are essential for capturing semantic inconsistencies. In particular, `orbit_class` and the region fields (`src_region`, `dst_region`) are characteristic of Satellite-IoT systems and describe orbital topology and spatial separation, thereby allowing the model to learn contextual constraints inherent to space-ground communication.

Unlike conventional IDSs based on static encodings or shallow features, our method enables semantic inference through a fine-tuned LLM while maintaining compatibility with resource-constrained nodes. The complete detection pipeline is illustrated in Fig. 1, and each component is explained in the following subsections.

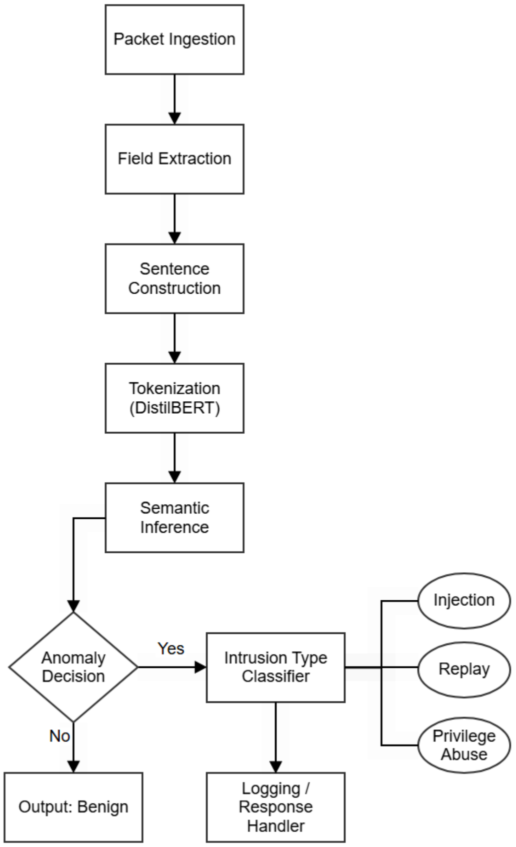

Fig. 1. End-to-end flow diagram of the proposed sentence-based intrusion detection process.

A. Packet Structure and Input Sentence Construction

Fig. 1 illustrates the end-to-end architecture of the proposed semantic anomaly detection approach. The approach comprises three main stages: (1) structured packet parsing and conversion into sentence-based representations, (2) semantic inference and anomaly classification using DistilBERT, and (3) attack-type categorization and logging. This design aims to preserve contextual semantics across heterogeneous communication flows among spaceborne nodes, ground control systems, and user-operated devices.

(1) Semantic Packet Parsing: The first stage transforms structured packet data into natural-language inputs suitable for DistilBERT-based models. It consists of three components: packet ingestion, field extraction, and sentence construction.

a) **Packet Ingestion**: Communication flows are generated based on scenario-specific constraints that reflect realistic interactions among ground (e.g., `gcs`, `gw1`), space (e.g., `leo`, `meo`), and user-segment nodes (e.g., `iot`, `uav`). Each link adheres to protocol-level constraints derived from CSP, MIOTY,

TABLE I
Structure and Description of Packet Fields in the Satellite-IoT Dataset

Field Name	Description	Example Values
timestamp	Packet generation time in ISO 8601 format	2025-07-01T03:15:20
src, dst	Valid communication nodes defined in NORMAL_LINKS	gw1→iot, leo→gcs[a]
priority	Message priority determined by msg_type	LOW, MEDIUM, HIGH, CRITICAL
src_port, dst_port	Ports assigned per node from SRC_PORT_MAP / DST_PORT_MAP	src=gw1→1883, dst=leo→3001[b]
src_region, dst_region	Regional codes derived from REGION_MAP	AS→AF, EQ→SP
orbit_class	Orbit category derived from ORBIT_CLASS_MAP	LEO, MEO, N/A
msg_type	Message type defined in VALID_MSG_TYPES per src–dst pair	telemetry, data, command, status, ack, alert[c]
payload_type	Field type determined by msg_type	TEMP, SIZE, MOVE, SIGNAL_LOSS, NORMAL[d]
payload	Formatted content generated according to payload_type	TEMP=24.5, command=RESET, ...
label	Class label used for anomaly detection	Normal, Injection, Replay, Privilege Abuse
ttl	Time-to-live value determined by src/dst role	64, 128, 200, 255
flags	Control flags associated with msg_type	ACK, PSH, SYN, ENC, RST[e]

[a] Examples include (gw1, iot), (rt, gw2), and (gcs, meo), based on NORMAL_LINKS.
[b] Port values are statically mapped per node; e.g., gw1:1883, leo:3001.
[c] Allowed message types are predefined for each src–dst pair in the system design.
[d] Mapping: telemetry→{TEMP, HUM, POS, BATT}, data→{COORD, SIZE, DATA_TYPE, REF_ID}, command→{ACTIVATE, MOVE, RESET, ...}, ack→{RECEIVED, EXECUTED}, alert→{ANOMALY_DETECTED, ...}, status→{NORMAL, LOW_BATTERY, ...}.
[e] Flag options include ACK, PSH, ENC, SYN, RST, etc., as defined in FLAGS_BY_MSGTYPE.

CCSDS, and TON_IoT specifications. For instance, telemetry messages are periodically broadcast from leo to ground nodes, whereas command transmissions originate exclusively from control nodes such as gcs to low-privilege endpoints like iot, thereby enforcing access hierarchies defined by mission protocols.

b) **Field Extraction**: Packets are parsed into 15 discrete fields encoding spatial, temporal, and logical metadata. These include routing identifiers (src, dst, src_port, dst_port), semantic fields (msg_type, payload_type, payload), timing information (timestamp, ttl), and control descriptors (priority, flags). Protocol-aware validation is applied to detect semantic inconsistencies. For example, a command packet with flag SYN from a sensor-class node (iot) to a control node (gcs) violates privilege boundaries and is classified as anomalous.

To ensure data validity, all synthesized packets were verified through range and type constraints (e.g., nonnegative ttl, predefined port sets, and valid msg_type–payload_type pairs). Normal packets were generated strictly within these protocol-defined limits, whereas attack samples intentionally violated one or more constraints—for instance, a field value exceeding the valid range (e.g., ttl=300 for a normal range of 0–255) or an invalid combination of msg_type and payload_type. For robustness evaluation, certain fields were intentionally omitted to simulate incomplete or lossy packet conditions. In such cases, the field label was retained in the input sentence, but its value was left blank, preserving the syntactic structure without injecting artificial tokens. For malformed numerical values, such as negative ttl, a sentinel indicator (e.g., INVALID_TTL) was used. This design allows the model to learn semantic irregularities arising from both missing and invalid fields, including those that explicitly violate protocol-level constraints, without explicit feature removal or imputation.

c) **Sentence Construction**: Validated fields are serialized into structured sentences that preserve semantic dependencies. A representative example is: *“At timestamp 2025-07-10T08:30:00Z, node iot (region=AF, orbit=LEO, port=3001) sent a command message to gcs (region=EU, port=1001) containing payload command=RESET, with priority HIGH, flag SYN, TTL 128, and protocol fields indicating payload_type=command.”*

This representation captures both field-level syntax and contextual intent, enabling the LLM to infer anomalies arising from spoofed roles, improper field combinations, or orbit–region mismatches. Unlike conventional IDSs that rely on flat feature vectors or static field encodings, our sentence-based formulation captures the semantic interplay among protocol fields—such as source role, message intent, and payload content—without requiring explicit parsers or manual feature design. This enables the LLM to infer context-sensitive anomalies, including unauthorized command injections, repeated telemetry sequences, and violations of communication hierarchies, which are typically indistinguishable in conventional field-isolated models.

(2) Semantic Inference and Anomaly Classification: After sentence construction, each packet sentence is processed through a lightweight semantic encoder based on DistilBERT. As illustrated in Fig. 1, this stage comprises three sequential modules: tokenization, semantic inference, and anomaly decision.

a) **Tokenization (DistilBERT):** Each serialized packet sentence is tokenized using the WordPiece tokenizer embedded in the DistilBERT architecture. The tokenizer decomposes domain-specific field–value pairs (e.g., command=RESET, orbit=LEO) into subword units, thereby enabling the model to capture fine-grained semantics of protocol terms. Special classification tokens—[CLS] for sentence-level summarization and [SEP] for boundary demarcation—are inserted to conform to the Transformer input format. The resulting token sequence is then mapped into high-dimensional embeddings that preserve both syntactic structure and contextual semantics, thereby optimizing performance under bandwidth-constrained and latency-sensitive aerospace environments. Tokenization follows the WordPiece algorithm inherited from BERT [23]. This subword-based method decomposes rare or compound identifiers into smaller semantic units, allowing consistent representation of protocol-specific tokens such as `orbit_class=LEO` or `payload_type=command`. WordPiece is particularly suitable for Satellite-IoT packet sentences, as it enables robust handling of unseen abbreviations and structured field values without expanding the vocabulary size.

b) **Semantic Inference:** The token embeddings are propagated through six Transformer encoder layers, each employing self-attention mechanisms and residual feed-forward operations. These layers learn long-range dependencies across heterogeneous fields such as `src_region`, `msg_type`, and `flags`, thereby enabling the model to infer higher-order semantic inconsistencies that span multiple fields. For instance, a `RESET` command transmitted from a low-trust node (e.g., `iot`) to a mission-critical control node (e.g., `gcs`), accompanied by a `SYN` flag, may indicate a potential privilege-escalation attempt. The attention weights dynamically adapt to cross-field interactions, thereby modeling latent security policies and protocol-specific role constraints without relying on protocol parsers or handcrafted logic.

The DistilBERT backbone employs six Transformer encoder layers distilled from BERT-base, providing a balanced trade-off between representational depth and computational efficiency [13]. Each layer refines token embeddings through multi-head self-attention and feed-forward transformations, enabling progressive modeling of inter-field dependencies across sentence-level packet representations. This architecture captures long-range contextual relations while remaining lightweight enough for onboard or gateway-level deployment.

c) **Anomaly Decision**: The final `[CLS]` token output represents a global semantic summary of the entire packet. This representation is passed through a lightweight classification head, which consists of a fully connected layer followed by a softmax (or sigmoid) activation. The head produces a scalar anomaly score, denoted as s_{anom}, which is compared against a threshold τ_{th} calibrated on the validation set. If $s_{\text{anom}} > \tau_{\text{th}}$, the packet is flagged as anomalous. This step remains computationally efficient and thus suitable for real-time inference on embedded systems. Moreover, the architecture is readily extensible to multi-class threat classification (e.g., injection, replay, privilege abuse) by adjusting the output dimension of the classifier.

This DistilBERT-based inference module enables robust detection of semantic-aware threats—such as injection, replay, and privilege abuse—even under partially degraded or missing-field conditions. Unlike conventional rule-based or statistical models that rely on rigid packet structures, the proposed approach generalizes across heterogeneous protocols without manual feature engineering and remains robust under bandwidth-limited satellite uplinks.

(3) Attack Type Classification and Logging: For packets identified as anomalous, a secondary classification module assigns a specific threat label—namely, *injection*, *replay*, or *privilege abuse*. These categories correspond to protocol-specific violations observed in Satellite-IoT environments and are derived from semantic inconsistencies detected during preceding encoding stages.

a) **Intrusion Type Classifier**: The final `[CLS]` token embedding, representing a holistic summary of the input sentence, is passed through a softmax-based multi-class classifier. This classifier is trained to differentiate intrusion types based on inter-field semantic patterns—such as mismatched payload values and flag combinations (injection), reused timestamp–payload pairs (replay), or unauthorized command issuance from low-trust nodes (privilege abuse). Each class label corresponds to a distinct protocol misuse scenario verified during dataset construction.

b) **Logging and Response**: Once classified, the threat label is appended to a log entry along with relevant metadata, including timestamp, source, destination, message type, and payload content. This audit trail facilitates post-event forensic analysis and enables policy-driven responses such as selective packet dropping, alert generation, or patch recommendations. In future implementations, these logs may also provide feedback for reinforcement learning–based adaptation or for retraining classifiers under evolving threat conditions.

This third stage completes the anomaly detection pipeline by producing explainable threat labels and structured logs. When combined with the preceding semantic parsing and binary anomaly detection stages, it forms a lightweight, semantic-aware intrusion detection mechanism tailored to the resource constraints of Satellite-IoT systems.

B. DistilBERT-Based Semantic Anomaly Classification

The proposed anomaly detection approach operates as an end-to-end semantic classification process tailored for structured Satellite-IoT packet streams. Unlike traditional modular intrusion detection systems that handle packet parsing, inference, and decision-making as separate stages, our architecture integrates protocol-aware sentence generation with Transformer-based embedding, thereby enabling holistic, semantic-aware threat classification. The detection pipeline comprises four key stages: (i) structured packet ingestion and sentence-based representation, (ii) semantic embedding via a lightweight Transformer encoder (DistilBERT), (iii) classification into operationally defined threat categories, and (iv) structured metadata logging for forensic traceability and downstream responses.

This architecture is designed to preserve semantic consistency across spatial, temporal, and logical fields while maintaining robust detection under incomplete, noisy, or protocol-divergent conditions. Each packet is represented as a natural-language sentence that captures field-level semantics, including orbit-region associations, privilege constraints conditioned on message type, and consistency between flags and payloads. The sentence is tokenized using the WordPiece tokenizer, encoded through six Transformer encoder layers, and summarized by the final `[CLS]` token embedding. The resulting sentence embedding is then passed through a fully connected classification head, which is trained in a supervised manner to assign an anomaly label.

To reflect security-relevant behavior in Satellite-IoT environments, the classifier predicts one of four semantic labels:

- **Normal:** The packet adheres to system-defined field constraints and segment-role policies. All field values—including source, destination, message type, and payload—comply with authorized communication flows.
- **Injection:** The packet contains malformed or semantically inconsistent payloads that violate structural rules or protocol specifications. These anomalies typically arise from unauthorized field manipulation or parser-directed attacks (e.g., `iot` → `gw1` with `msg_type=command`, `payload=CALIBRATE`, `flag=URG`).
- **Replay:** The packet reuses previously transmitted timestamp and payload values, thereby violating temporal consistency. Such behavior may lead to state inconsistencies or trigger unintended operations (e.g., repeated telemetry from `uav` → `gw2` with identical timestamp and `payload=POS=37.4,127.1`).
- **Privilege Abuse:** A low-privilege node (e.g., `iot`) attempts to issue control-level commands (e.g., `RESET`, `SHUTDOWN`) to high-privilege nodes (e.g., `gcs`), thereby violating access-control policies enforced within the communication model (e.g., `iot` → `gcs` with `command=SHUTDOWN`).

This threat-aware semantic classification enables the model to detect anomalies that, while structurally valid, violate contextual expectations. For instance, a packet containing `command=RESET` transmitted from an `iot` node may conform to protocol syntax yet violate sender-privilege policies. Likewise, a telemetry message exchanged between disallowed node pairs may appear syntactically correct yet raise behavioral concerns. By modeling inter-field relationships through self-attention mechanisms, the system performs reliably without handcrafted rules.

Each threat class represents a distinct security risk within Satellite-IoT systems: *injection* denotes payload-integrity violations, *replay* concerns the freshness and correctness of temporal sequences, and *privilege abuse* involves unauthorized attempts to execute privileged operations. This mapping to domain-relevant threat semantics enhances both interpretability and operational relevance, ensuring applicability in real-world environments. In addition, these categories collectively capture a broad spectrum of semantic inconsistencies observable at the packet level, ranging from data-field manipulation to cross-segment privilege escalation. By associating each anomaly type with a specific operational context, the classification results can be directly interpreted in terms of system behavior and mission impact, thereby bridging the gap between packet-level detection and system-level situational awareness.

Section IV presents an empirical evaluation of the proposed detection approach under realistic Satellite-IoT conditions, encompassing both normal and adversarial scenarios. The model's performance is compared with existing intrusion detection methods—including rule-based and learning-based approaches—to evaluate robustness and semantic generalization.

IV. Experimental Results and Discussion

This section presents a comprehensive evaluation of the proposed DistilBERT-based semantic anomaly detection approach.

A. Experimental Environment and Dataset Construction

We utilize a pre-trained DistilBERT model from Hugging Face, fine-tuned for sentence-level multi-class classification. Each input sentence is derived from a structured packet comprising 15 fields, including attributes such as `src`, `dst`, `msg_type`, `orbit_class`, and `payload`. A representative input sentence (partial view) is as follows: *"Telemetry message from `leo` to `gcs` carrying `TEMP=22.5` with priority `HIGH` and flag `ENC` at timestamp `2025-07-10T08:30:00Z`."* To illustrate an anomalous case, consider: *"At timestamp `2025-07-10T08:45:12Z`, node `iot` (region=`AF`, orbit=`LEO`) transmitted a `command=RESET` message to `gcs` with priority `HIGH` and flag `SYN`, thereby violating access-control constraints."*

The objective of this evaluation is to assess the model's capability to classify Satellite-IoT packets into four security-relevant categories—*normal*, *injection*, *replay*, and *privilege abuse*—through three complementary experiments: (i) performance benchmarking against baseline models, (ii) evaluation under scenario-specific anomalies, and (iii) interpretability analysis via attention-based field attribution.

To support these experiments, a scenario-driven Satellite-IoT dataset was constructed using protocol-constrained synthesis. The dataset integrates packet structures and field characteristics derived from CSP logs, MIOTY sensor traces collected in our testbed, CCSDS-formatted messages, and traffic patterns from the TON_IoT dataset. These heterogeneous sources were analyzed to establish unified specifications representing realistic cross-segment communication among space, ground, and user nodes. Custom Python scripts were employed to synthesize packets according to these specifications, generating both normal and malicious traffic through controlled randomization and targeted field manipulation corresponding to *injection*, *replay*, and *privilege abuse* behaviors. This synthetic generation ensures reproducible experimentation while avoiding the use of sensitive real-world operational data.

The training set consists of 25,000 samples—comprising 10,000 normal packets and 15,000 attack instances equally distributed across *injection*, *replay*, and *privilege abuse*. For evaluation, 5,000 fully structured packets were used as the primary test set. In addition, 5,000 test samples with randomly missing two to five fields were constructed to assess robustness under incomplete input conditions, simulating data degradation in lossy Satellite-IoT environments. All experiments are conducted using fixed training and test splits derived from the structured packet dataset. The model is trained using the AdamW optimizer for ten epochs, with a batch size of 16 and a maximum sequence length of 128. To address class imbalance, a weighted cross-entropy loss function is employed. Evaluation metrics include overall accuracy and weighted F1-score.

B. Computational Efficiency and Edge Feasibility

We adopt DistilBERT as the base model owing to its efficiency and suitability for deployment on resource-constrained systems. With approximately 66 million parameters, it offers a lightweight alternative to BERT-base while maintaining strong detection capability [13]. To quantify the computational cost of the proposed approach, two primary metrics were measured. All training and inference experiments were conducted on a desktop running Windows 11 (64-bit) equipped with an Intel Core i7-11700 CPU, 64 GB of system memory, and an NVIDIA GeForce RTX 3060 GPU with 12 GB of VRAM. The implementation was developed in Python 3.11 using PyTorch v2.7.1 (CUDA-enabled) and the Hugging Face Transformers library (v4.53.0). The pre-trained model used in all experiments was `distilbert-base-uncased` from the Hugging Face model repository. Training was performed for 10 epochs with a batch size of 16 and a maximum sequence length of 128. The AdamW optimizer—the default optimization method for Transformer-based models—was employed for fine-tuning due to its effective weight-decay regularization and stable convergence properties [24]. Under this environment, the observed peak GPU memory consumption during training (batch size = 16) reached approximately 789 MB, and the average inference latency per packet was 26.2 ms.

The 789 MB figure corresponds to the GPU-based training configuration, which includes additional memory allocations for optimizer states and backpropagation buffers. During training, extra memory is reserved for gradient computation and optimizer updates, whereas these components are absent during inference. In operational scenarios, model training is performed offline, and only the fine-tuned DistilBERT weights are executed for inference on satellite or ground-segment devices. Under inference-only conditions, gradients are disabled, and mixed-precision execution can be applied, substantially reducing memory demand. The inference footprint is primarily determined by model parameters (approximately 66 million, typically stored in 250–300 MB) and minimal runtime activations. Preliminary validation on two edge-class platforms—a Raspberry Pi 4B (8 GB RAM) connected to MIOTY sensor nodes and an NVIDIA Jetson Nano used for embedded inference—confirmed that the model consistently operated within the 250–300 MB memory range, with minor variations depending on runtime configurations. These results demonstrate the feasibility of deploying the proposed model on compact edge hardware in Satellite-IoT environments.

Compared with larger Transformer models such as BERT-base, which typically require more than 1.2 GB of memory even for inference [25], DistilBERT provides a lightweight yet semantically capable alternative suitable for near real-time packet-level analysis in Satellite-IoT systems [26]. Validation of performance and precise resource utilization within an operational satellite environment remains an important direction for future work. The evaluation encompasses both standard classification performance under balanced label distributions and manually constructed, scenario-based anomalies that reflect realistic security threats. These include malformed payloads, unauthorized command issuance, and timestamp–payload duplication indicative of replay attacks. The following subsections present detailed evaluation results across three dimensions: baseline comparison, scenario-specific detection accuracy, and attention-based interpretability.

TABLE II
PERFORMANCE COMPARISON OF DETECTION MODELS ON SATELLITE–IOT PACKET CLASSIFICATION.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Snort	48.1	48.0	39.0	34.0
Random Forest	87.6	88.0	87.0	87.0
LSTM	92.8	95.0	91.0	92.3
DistilBERT	99.0	99.0	99.0	98.9

C. Performance Comparison with Baseline Models

We evaluate the effectiveness of the proposed semantic anomaly detection approach by comparing it with three representative baselines: a rule-based engine (Snort [27]), a traditional machine-learning classifier (Random Forest [28]), and a sequence-aware deep-learning model (LSTM [29]). The results demonstrate that sentence-level modeling of inter-field semantics significantly enhances anomaly detection accuracy compared with conventional approaches that process each packet field independently without contextual integration.

As shown in Table II, the rule-based Snort engine exhibits low recall and F1-score due to its reliance on static signatures, which renders it ineffective against semantically or structurally inconsistent packets. The Random Forest classifier benefits from data-driven learning but lacks awareness of semantic relationships among protocol fields, resulting in only moderate performance. The LSTM model demonstrates improved sensitivity by capturing sequential dependencies; however, its limited ability to model non-local interactions—such as inconsistencies between flags and payloads or mismatches between region and role—reduces its robustness in multi-field anomaly scenarios. On the test dataset, the proposed DistilBERT-based approach outperforms all baseline methods across every metric, achieving 99.0% accuracy and 98.9% F1-score. This performance improvement is attributed to the ability of DistilBERT to encode sentence-level representations of packet structures and capture inter-field dependencies via self-attention mechanisms. It accurately detects semantic anomalies that span multiple fields, such as a `RESET` command issued by a non-authoritative node or a telemetry message with a contextually valid payload but paired with an unexpected or unauthorized flag.

These results demonstrate that Transformer-based semantic modeling can effectively capture inter-field dependencies and

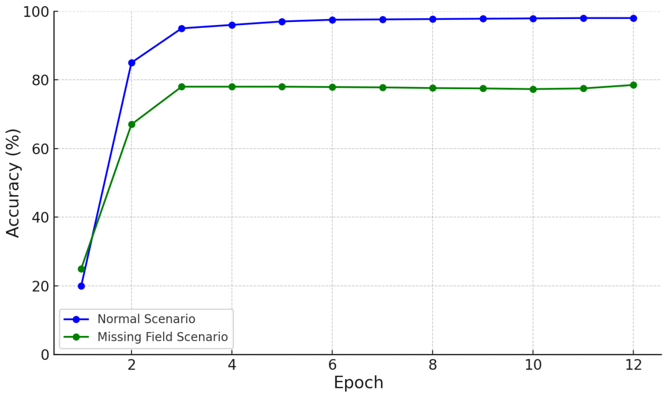

Fig. 2. Validation accuracy trends across epochs for normal and missing-field scenarios in Satellite–IoT packet classification.

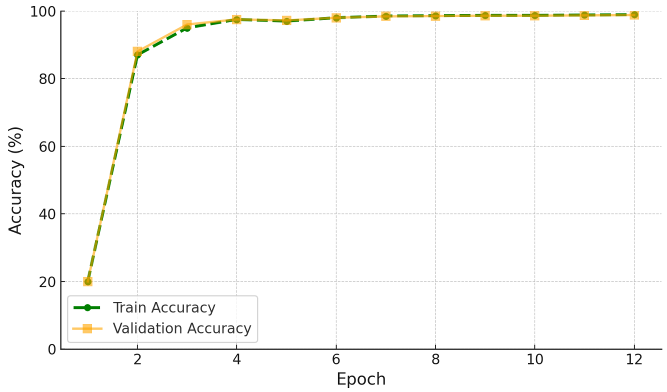

Fig. 3. Training and validation accuracy per epoch under normal packet conditions.

contextual inconsistencies. Such models are particularly suitable for the Satellite-IoT domain, where attacks often involve subtle semantic inconsistencies rather than explicit field-level anomalies [9].

D. Evaluation on Scenario-Based Anomalies

We evaluate the model's robustness against anomalies that reflect realistic operational challenges in Satellite-IoT networks, including partial field omissions, inconsistent payload–flag combinations, and unauthorized command injections. We consider two scenarios: (i) **Normal packets**, which include all 15 predefined fields, and (ii) **Missing-field packets**, in which two or more noncritical fields—such as flags, orbit class, or time-to-live—are randomly omitted. These scenarios emulate anomalous telemetry resulting from transmission faults, protocol mismatches, or adversarial field tampering. This evaluation assesses the model's ability to generalize its semantic reasoning to incomplete inputs, even when trained exclusively on fully structured packets. This generalization capability is critical for practical deployment in Satellite-IoT networks, where intermittent connectivity and resource limitations frequently result in partial field dropout.

Figure 2 illustrates the validation accuracy across epochs for both input scenarios. Under normal conditions, the model converges rapidly, achieving over 99% accuracy by the fourth epoch. This indicates strong internalization of inter-field dependencies when the full packet structure is available. In contrast, when key fields are missing, the model maintains a consistent validation accuracy of approximately 78%, demonstrating tolerance to incomplete input. Despite the absence of key semantic cues such as `msg_type`, `payload`, and `src`, the model achieves robust classification performance. Traditional ML and noncontextual DL baselines, on the other hand, either fail to converge during training or exhibit near-random classification under the same conditions, often unable to detect any anomaly. This contrast underscores the advantage of our attention-based approach in handling degraded or partially structured packets by inferring missing semantics from surrounding context. Furthermore, to assess training stability, Figure 3 compares training and validation accuracy under normal packet conditions. The close alignment of the two curves confirms effective convergence without overfitting, indicating that the model generalizes well even when trained solely on fully structured inputs.

These results demonstrate the practical viability of the proposed model in Satellite-IoT environments, showing its ability to generalize effectively even when packet inputs are incomplete or partially corrupted due to bandwidth limitations, transmission faults, or adversarial tampering.

E. Analysis of Attention-Based Feature Importance

Interpretability in decision-making is essential when deploying anomaly detection systems in security-sensitive environments such as Satellite-IoT networks, where operational transparency is critical for trust and accountability. To evaluate the interpretability of the proposed model, we analyze attention weights extracted from the final Transformer encoder layer of DistilBERT. This layer captures high-level dependencies across structured packet fields, providing insight into how the model internally represents and prioritizes threat-relevant information.

Table III summarizes the average attention allocation across all 15 structured fields, categorized by the four threat labels: normal, injection, replay, and privilege abuse. The results reveal distinct and consistent patterns of attention distribution aligned with the semantic properties of each threat class. For instance, the `flags` field receives the highest attention in both the normal and injection classes, suggesting that control-level indicators such as `ACK` and `URG` are central to distinguishing benign traffic from tampered packets—particularly when payload structure alone is ambiguous or unaltered. In the replay class, the model assigns dominant attention weight to the `timestamp` field, reflecting its reliance on temporal redundancy as a key anomaly indicator. In contrast, for privilege abuse, attention concentrates on the `src`, `src_port`, and `priority` fields, indicating that the model captures role-based access violations through origin identity and command criticality.

TABLE III
AVERAGE ATTENTION WEIGHTS PER FIELD ACROSS FOUR CLASSIFICATION LABELS

Field Name	Normal	Injection	Privilege Abuse	Replay
flags	**0.0231**	**0.0140**	0.0091	**0.0097**
timestamp	0.0060	0.0102	**0.0105**	**0.0088**
src	0.0080	**0.0103**	0.0098	0.0078
src_region	**0.0092**	0.0087	0.0091	0.0084
payload	0.0049	0.0080	0.0087	0.0083
priority	0.0037	0.0073	**0.0103**	0.0081
orbit_class	0.0049	0.0071	0.0091	0.0076
dst_region	0.0072	0.0069	0.0087	0.0078
src_port	0.0048	0.0070	0.0086	0.0078
msg_type	0.0074	0.0068	0.0082	0.0083
payload_type	0.0032	0.0052	0.0082	0.0079
dst_port	0.0033	0.0057	0.0083	0.0075
ttl	0.0027	0.0050	0.0084	0.0066
dst	0.0025	0.0052	0.0086	0.0062

These observations indicate that the model does not rely on a fixed set of predefined features but instead dynamically adjusts its attention depending on the context and semantics of each threat type. In contrast to traditional IDS approaches that treat packet fields independently or rely on rigid rule sets, the Transformer-based design captures meaningful interactions across fields that reflect abnormal protocol behavior. This flexible attention mechanism enhances generalization to unfamiliar patterns while offering interpretability at the field level—thereby facilitating understanding of how specific combinations of inputs lead to each decision. By aligning attention with threat-specific characteristics, the model helps translate structural input variations into clear, security-relevant signals for Satellite-IoT applications.

V. CONCLUSION AND FUTURE WORK

In this paper, we proposed a semantic anomaly detection approach for Satellite-IoT networks that leverages sentence-based representations of structured packets and a lightweight Transformer model, DistilBERT. Each packet, composed of 15 predefined fields, was converted into natural-language-like sentences to enable contextual reasoning and cross-field dependency modeling—key capabilities for identifying protocol-level anomalies in space communication systems. Experimental evaluation across four categories (normal, injection, replay, and privilege abuse) showed that our DistilBERT-based model outperformed rule-based systems (Snort), classical machine learning (Random Forest), and sequential deep learning models (LSTM), achieving 99.0% accuracy and 98.9% F1-score with low inference latency and memory usage. Scenario-based evaluation confirmed robustness under missing-field conditions, while attention-based interpretation showed that the model's field-level focus aligns with domain-specific threat semantics.

While effective for single-packet anomalies, the current approach does not explicitly model temporal dependencies across sequential packets; as a result, certain false positives or missed detections may occur when session-level context is required. Future work will therefore extend the framework to multi-packet and time-series inference to incorporate temporal context for more reliable session-level detection. Another direction is integrating LLM-based anomaly detection with hierarchical threat modeling frameworks [2], [30], abstracting packet-level predictions into higher-level semantic representations for system-wide threat propagation analysis. To enhance interpretability and operational alignment, we will also explore tactics, techniques, and procedures (TTP)-based abstraction that links detected anomalies with adversarial tactics in structured threat models [31]. Finally, we will broaden coverage to cross-layer threats at the physical and MAC layers (e.g., spoofing, signal manipulation, jamming) and validate computational feasibility on onboard processors to assess operational constraints.

In conclusion, the proposed method provides a practical and interpretable framework for semantic anomaly detection in Satellite-IoT environments, combining structured packet modeling with lightweight Transformer-based inference to enhance both detection accuracy and system resilience.

ACKNOWLEDGMENT

This work was supported by the Korea Institute for Advancement of Technology (KIAT) grant funded by the Ministry of Trade, Industry and Energy (MOTIE), Korea Government (No. RS-2021-KI002499, HRD Program for Industrial Innovation).

REFERENCES

[1] S. Salim, N. Moustafa, and M. Reisslein, "Cybersecurity of satellite communications systems: A comprehensive survey of the space, ground, and links segments," IEEE Commun. Surveys Tuts., vol. 27, no. 1, pp. 372–425, 2025.

[2] J. Park, T. Eom, H. Kim, H. Park, Z. Yoon, and J. Park, "Threat vector–hierarchical attack representation model-based threat modeling and security assessment for satellite networks," Appl. Sci., vol. 15, no. 5, Art. no. 2751, 2025. [Online]. Available: https://doi.org/10.3390/app15052751

[3] Unit 42, "IoT adoption and its security risks have both grown," Palo Alto Networks, 2020. [Online]. Available: https://start.paloaltonetworks.com/unit-42-iot-threat-report

[4] K. Prabu, P. Sudhakar, T. Manikandan, B. Balusamy, and F. Benedetto, "A novel hybrid unsupervised learning approach for enhanced cybersecurity in the IoT," Future Internet, vol. 16, no. 7, p. 253, 2024. [Online]. Available: https://doi.org/10.3390/fi16070253

[5] A. Alsaedi, N. Moustafa, B. Turnbull, and K.-K. R. Choo, "TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems," IEEE Access, vol. 8, pp. 165130–165150, 2020. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.3022862

[6] N. Boschetti, N. Gordon, and G. Falco, "Space cybersecurity lessons learned from the Viasat cyberattack," in Proc. ASCEND Conf., Las Vegas, NV, USA, Oct. 2022.

[7] M. Manulis, C. P. Bridges, R. Harrison, V. Sekar, and A. Davis, "Cyber security in New Space: Analysis of threats, key enabling technologies and challenges," Int. J. Inf. Secur., vol. 20, no. 3, pp. 287–311, Jun. 2021. [Online]. Available: https://link.springer.com/10.1007/s10207-020-00503-w

[8] O. Driouch, S. Bah, and Z. Guennoun, "Distributed intrusion detection system for CubeSats, based on deep learning packets classification model," 2024. [Online]. Available: https://doi.org/10.23919/3s60530.2024.10592294

[9] M. Hassanin *et al.*, "PLLM-CS: Pre-trained large language model (LLM) for cyber threat detection in satellite networks," Ad Hoc Netw., vol. 166, p. 103645, 2024. [Online]. Available: https://doi.org/10.1016/j.adhoc.2024.103645

[10] A. Aldhaheri, F. Alwahedi, M. A. Ferrag, and A. A. Battah, “Deep learning for cyber threat detection in IoT networks: A review,” Internet Things Cyber-Phys. Syst., 2023. [Online]. Available: https://doi.org/10.1016/j.iotcps.2023.09.003

[11] J. L. Delgado and J. A. Ramos, “A comprehensive survey on generative AI solutions in IoT security,” Electronics, vol. 13, no. 24, p. 4965, 2024. [Online]. Available: https://doi.org/10.3390/electronics13244965

[12] A. Guma *et al.*, “Securing the Internet of Wetland Things (IoWT) using machine and deep learning methods: A survey,” Mesopotamian J. Comput. Sci., vol. 2025, pp. 17–63, 2025. [Online]. Available: https://doi.org/10.58496/MJCSC/2025/002

[13] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter,” arXiv preprint arXiv:1910.01108, Oct. 2019. [Online]. Available: https://arxiv.org/abs/1910.01108

[14] CSP Project, “The CubeSat Space Protocol,” 2023. [Online]. Available: https://libcsp.github.io/libcsp/

[15] MIOTY Alliance, “MIOTY protocol specification,” 2021. [Online]. Available: https://mioty-alliance.com/technology/

[16] Consultative Committee for Space Data Systems, “Telemetry channel coding,” CCSDS 101.0-B-7, Sep. 2020. [Online]. Available: https://public.ccsds.org/Pubs/101x0b7.pdf

[17] J. Wang, H. Li, L. Wang, and Z. Xu, “Satellite telemetry data anomaly detection using multiple factors and co-attention based LSTM,” in Proc. IEEE Wireless Commun. Netw. Conf. (WCNC), Glasgow, U.K., Mar. 2023, pp. 1–6. [Online]. Available: https://doi.org/10.1109/WCNC55385.2023.10118903

[18] O. Driouch, S. Bah, and Z. Guennoun, “Intrusion detection system for CubeSats: A survey,” in Proc. Int. Wireless Commun. Mobile Comput. Conf. (IWCMC), Jun. 2023, pp. 596–601.

[19] A. Aldhaheri, F. Alwahedi, M. A. Ferrag, and A. A. Battah, “Deep learning for cyber threat detection in IoT networks: A review,” Internet Things Cyber-Phys. Syst., 2023. [Online]. Available: https://doi.org/10.1016/j.iotcps.2023.09.003

[20] Y. Chen *et al.*, “A survey of large language models for cyber threat detection,” Comput. Secur., 2024. [Online]. Available: https://doi.org/10.1016/j.cose.2024.104016

[21] M. A. Ferrag *et al.*, “Revolutionizing cyber threat detection with large language models: A privacy-preserving BERT-based lightweight model for IoT/IIoT devices,” IEEE Access, vol. 12, pp. 23733–23750, 2023. [Online]. Available: https://doi.org/10.1109/ACCESS.2024.3363469

[22] D. A. Worae, A. Sheikh, and S. Mastorakis, “A unified framework for semantic-aware IoT management and state-of-the-art IoT traffic anomaly detection,” 2024. [Online]. Available: https://doi.org/10.48550/arxiv.2412.19830

[23] J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol. (NAACL-HLT)*, Minneapolis, MN, USA, 2019, pp. 4171–4186. [Online]. Available: https://doi.org/10.18653/v1/N19-1423

[24] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019. [Online]. Available: https://openreview.net/forum?id=Bkg6RiCqY7

[25] C. Xiong, “A survey of transformer optimization techniques: Progress and challenges from computational efficiency to multimodal fusion,” Appl. Comput. Eng., vol. 157, pp. 139–146, Jul. 2025. [Online]. Available: https://doi.org/10.54254/2755-2721/2025.PO24682

[26] X. Nie, “Optimizing transformer models for edge deployment in autonomous vehicles: Lightweight architectures and quantization strategies for embedded vision,” [Journal name missing], vol. 2025, pp. 36–50, Jun. 2025.

[27] M. Roesch, “Snort: The open source network intrusion detection system,” 1998. [Online]. Available: https://www.snort.org

[28] L. Breiman, “Random forests,” Mach. Learn., vol. 45, pp. 5–32, 2001. [Online]. Available: https://doi.org/10.1023/A:1010933404324

[29] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Comput., vol. 9, pp. 1735–1780, 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[30] S. Tete, “Threat modelling and risk analysis for large language model (LLM)-powered applications,” 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2406.11007

[31] Y. Sharma, S. Birnbach, and I. Martinovic, “RADAR: A TTP-based extensible, explainable, and effective system for network traffic analysis and malware detection,” in Proc. ACM Workshop Artif. Intell. Secur. (AISec), Jun. 2023, pp. 159–166. [Online]. Available: https://doi.org/10.1145/3590777.3590804