MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE

Belawadi, SrirangapatnaTq, Mandya-571477

DEPARTMENT OF CSE (Artificial Intelligence)

Assignment – Project Report

2025-26

Subject Name:Research Methodology and Intellectual Property Rights

Subject Code:M23BRMK507

Semester:5th

Submitted by: Team No:

Sl. No.	Student Name	USN.		CO's Mapping				Total	Scaled to
			CO1	CO2	CO3	CO4	CO5		
1	Sanketh H N	4MH23CA049							

Verified and Approved by:

Faculty Name:Prof.Ragavendra

Signature:

Date:

Project GitHub Repository:
Link:https://github.com/Sankethhn/Rm

TITLE OF INVENTION

“Adaptive AI for Real-Time Cyber Threat Defences in an AI-driven security system”

INVENTOR:

SANKETH H N

MAHARAJA INSTITUTE OF TECHNOLOGY, MYSORE

ABSTRACT

The present invention discloses an intelligent **network security infrastructure** designed for **autonomous real-time cyber threat detection and mitigation** at the **network edge**. Unlike conventional intrusion detection systems that rely on **static signatures** and **centralized analysis**, the proposed system employs a combination of **graph neural networks (GNNs)**, **long short-term memory (LSTM) networks**, and **reinforcement learning (RL)** to continuously model and adapt to evolving attack behaviors in **IP networks**. The system converts live packet streams into **temporal communication graphs**, learns **spatio-temporal patterns** of normal traffic, and identifies anomalies such as **zero-day exploits**, **distributed denial-of-service (DDoS) attacks**, and **advanced persistent threats (APTs)** in **sub-second timescales**. Upon detecting a high-risk anomaly, an **Adaptive Policy Orchestration Module** automatically selects and enforces context-aware countermeasures—such as **source quarantine**, **rate limiting**, **traffic rerouting**, or **honeypot redirection**—using **WebAssembly (WASM) security microservices** that can be deployed without interrupting live traffic. Furthermore, the system incorporates an **Explainable AI Engine** based on **SHAP attribution**, which generates human-readable justifications for each action, and a **Federated Learning Module** that allows multiple edge nodes to collaboratively improve their models without sharing raw packet data. The invention thus provides a **privacy-preserving**, **low-latency**, and **self-evolving** solution for enterprise networks, data centers, and critical infrastructures, offering robust defense against novel cyber attacks while maintaining operational transparency for security operators.

FIELD OF THE INVENTION

The present invention relates to **cybersecurity systems**, and more particularly to **real-time adaptive artificial intelligence systems** for detecting, predicting, and mitigating cyber threats at **network edge devices** using **graph neural networks**, **LSTM predictive models**, **reinforcement learning agents**, and **federated learning protocols**.

The invention further relates to **explainable artificial intelligence (xAI) methods** for cybersecurity, particularly **SHAP-based attribution techniques** for justifying automated threat response decisions, and to **WebAssembly-based microservices architectures** for deploying adaptive security policies without downtime on **edge computing devices** such as routers, firewalls, and network gateways.

The invention finds application in:

- Enterprise network security infrastructure
- Data centre threat detection and mitigation
- Critical infrastructure protection (power grids, industrial control systems, medical networks)
- Cloud edge computing security
- Telecommunication network defence

BACKGROUND OF THE INVENTION

Problem Statement

Traditional network intrusion detection systems (IDS) rely on **signature-based pattern matching** or **static anomaly thresholds**, limiting their effectiveness against:

- **Zero-day exploits**: Attacks employing previously unknown vulnerabilities that bypass signature databases
- **Distributed Denial of Service (DDoS) variants**: Rapidly evolving attack patterns that evade fixed detection rules
- **Advanced persistent threats (APTs)**: Multi-stage, adaptive attacks that evolve in real-time in response to defensive measures
- **Latency constraints**: Classical centralized security systems require packet forwarding to cloud centers, introducing **100–500ms delays** incompatible with high-frequency trading, industrial control systems, or real-time medical networks

Current state-of-the-art systems (e.g., Snort, Zeek, Suricata) achieve mitigation response times of **50–200ms**, insufficient for millisecond-critical applications. Additionally, most systems lack **interpretability**, making it difficult for security operators to understand why a particular threat was flagged or how automated responses were selected, raising organizational and regulatory compliance concerns.

Limitations of Prior Art

- **Static Graph Models** (US Patent 11,184,401 B2): Existing AI-driven cybersecurity systems model network traffic as static snapshots, missing temporal attack evolution patterns.
- **Centralized Federated Learning** (US Patent 12,093,837 B2): Privacy-preserving threat models require round-trip communication to cloud centers, increasing latency and reducing autonomous edge decision-making.
- **Black-Box RL Policies**: Reinforcement learning agents in security applications typically lack explainability, hindering operator trust and regulatory compliance.

SUMMARY OF THE INVENTION

Perception Layer

Real-time packet capture via **extended Berkeley Packet Filter (eBPF)** or **Scapy libraries** converts network traffic streams into **temporal property graphs**. Nodes represent network entities (source IP, destination IP, port, service protocol); edges encode flow properties (bytes, packet count, flags, inter-arrival times).

Reasoning Layer

A **hybrid deep learning architecture** processes graph data:

- **Graph Neural Network (GNN)**: Learns node embeddings that capture spatial patterns of network relationships and traffic characteristics.
- **LSTM Network**: Processes sequential GNN embeddings over time windows (e.g., 5-second sliding windows) to predict threat evolution.
- **Anomaly Scorer**: Computes anomaly likelihood via statistical divergence from baseline traffic profiles.
- **RL Policy Network**: Selects mitigation actions from a discrete action space (e.g., quarantine IP, insert honeypot, reroute traffic, escalate to SIEM).
- **Confidence Scorer**: Estimates decision confidence; low-confidence detections trigger human-in-loop escalation.

Action Layer

Selected mitigation actions are compiled into **WebAssembly (WASM) bytecode** and deployed to firewall/router data planes without service downtime. Executed actions include:

- IP quarantine via iptables/netfilter rules
- Honeypot traffic injection (decoys to engage attackers)
- Micro-segmentation rule insertion
- Traffic rerouting via BGP announcements or MPLS label switching

Explainability and Learning

- **SHAP Attribution**: For each anomaly detection, computes feature importance scores explaining which graph edges/nodes most influenced the threat decision.
- **Federated Model Sharing**: Edge devices securely share model parameter deltas (not raw traffic data) via differential privacy-protected channels, enabling collective threat intelligence without data leakage.
- **Human-in-Loop Escalation**: Alerts with confidence <70% are escalated to security operators with full SHAP explanations and recommended actions.

Key Technical Innovations

The system claims technical novelty in:

1. Dynamic property graph construction from packet streams for real-time threat modeling
2. Hybrid GNN+LSTM architecture for spatio-temporal threat detection
3. RL-based adaptive policy selection with confidence-scored human escalation
4. SHAP-based explainability for black-box threat decisions
5. WASM microservices for zero-downtime policy updates
6. Federated learning with differential privacy for collaborative edge defense

DETAILED DESCRIPTION OF THE INVENTION

System Architecture Overview

The system comprises five core modules as depicted in **Figure 1**:

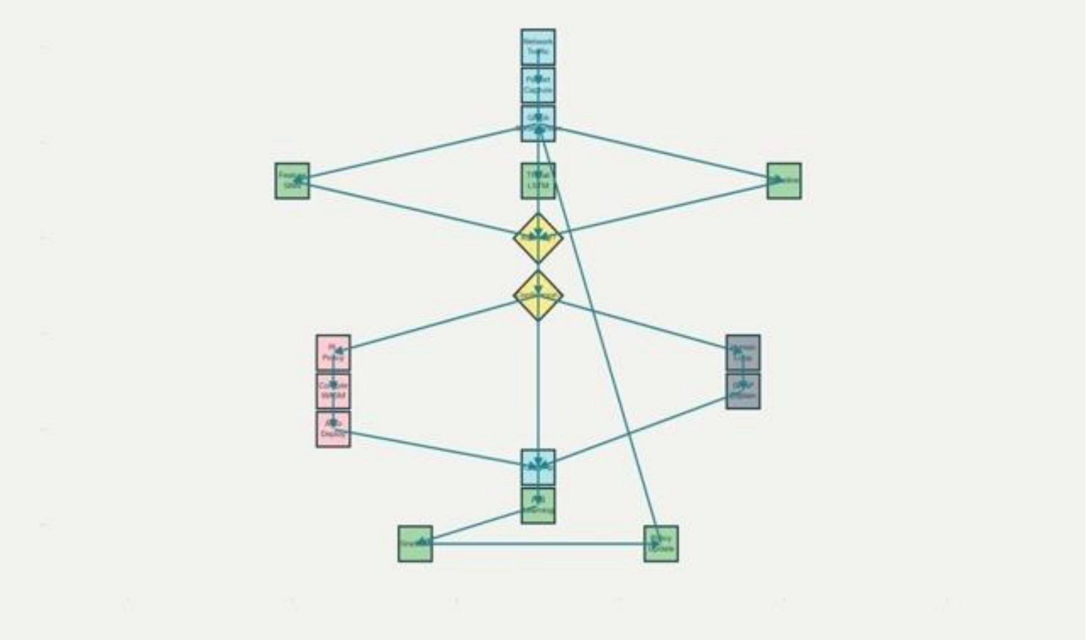

Fig-1: AI Cyber Threat Defense Flow

Traffic Capture and Graph Construction Module

Input: Raw network packets (captured via eBPF or tcpdump)

Process:

1. Parse packet headers (IP, TCP/UDP, payload)
2. Extract 5-tuple (src_ip, dst_ip, src_port, dst_port, protocol)
3. Group packets by 5-tuple into bidirectional flows

4. Compute flow statistics over sliding 5-second windows:
    - Total bytes transferred (upload, download)
    - Packet count and inter-arrival times
    - Protocol flags (SYN, FIN, RST, URG, PSH)
    - Payload entropy (to detect encrypted vs. cleartext traffic)
5. Construct property graph where:
    - Nodes: {IP addresses, ports, services}
    - Edges: Flow objects with timestamp, stats
    - Node attributes: Geographic location, WHOIS data, reputation score
    - Edge attributes: Bytes, packets, flags, entropy, TTL

Output: Temporal property graph $G_t = (V_t, E_t, A_t)$ at each window

Feature Extraction via GNN Module

Input: Property graphs $\{G_0, G_1, \ldots, G_T\}$

Algorithm: Message Passing Neural Network (MPNN)

1. Initialize node embeddings $h_v^{(0)}$ from node attributes
2. For $k = 1$ to K layers:
    - Aggregate messages from neighbors: $m_v = \text{AGGREGATE}(\{h_u^{(k-1)}: u \in \text{neighbors}(v)\})$
    - Update node embeddings: $h_v^{(k)} = \text{UPDATE}(h_v^{(k-1)}, m_v)$

3. For each edge (u, v), compute edge embedding: $e_{uv} = \text{CONCAT}(h_u^{(K)}, h_v^{(K)}, a_{uv})$

4. Compute graph-level embedding: $h_G = \text{READOUT}(\{e_{uv}, h_v^{(K)}\})$

5. Output: Sequence of graph embeddings $[h_G^1, h_G^2, \ldots, h_G^T]$

Baseline Profile Learning:

- Train GNN on benign traffic for 7 days to establish "normal" baseline embeddings
- Compute mean μ_{benign} and covariance Σ_{benign} of baseline embeddings

Threat Prediction via LSTM Module

Input: GNN embedding sequence $[z_1, z_2, \ldots, z_T]$ and baseline profile

Process:

1. Compute deviation: $\Delta z_t = z_t - \mu_{\text{benign}}$ (normalized via Σ_{benign})

2. Feed deviation sequence to LSTM:
   - LSTM processes temporal patterns in deviations
   - Output hidden states: $[h_1, h_2, \ldots, h_T]$

3. Compute anomaly score via statistical distance:
   - $a_t = \text{Mahalanobis}(z_t, \mu_{\text{benign}}, \Sigma_{\text{benign}})$
   - OR $a_t = \sigma(\text{Dense}(h_T))$

4. Predict threat probability:
   - threat_prob$_t$ = SOFTMAX(Dense(h_T))
   - Classes: {benign, reconnaissance, exploitation, exfiltration, C2}

Output: Anomaly score $a_t \in [0,1]$, predicted threat class y_t

Adaptive RL Policy Module

State: $s_t = \{\text{anomaly}\backslash\_\text{score}_t, \text{threat}\backslash\_\text{class}_t, \text{network}\backslash\_\text{topology}, \text{device}\backslash\_\text{resources}\}$

Action Space A:

- a_0: Continue monitoring (no action)
- a_1: Quarantine source IP (iptables DROP rule)
- a_2: Insert honeypot (create decoy service)
- a_3: Reroute traffic (MPLS label switching)
- a_4: Rate limit (token bucket algorithm)
- a_5: Escalate to SIEM (alert operator)

RL Algorithm: Proximal Policy Optimization (PPO)

Reward Function:

$$r(s,a) = \alpha \times (\text{threat}\backslash\_\text{mitigated}) - \beta \times (\text{false}\backslash\_\text{positive}\backslash\_\text{cost}) - \gamma \times (\text{action}\backslash\_\text{latency}\backslash\_\text{ms}/10) - \delta \times (\text{network}\backslash\_\text{disruption})$$

Where coefficients $\alpha, \beta, \gamma, \delta$ are tuned via simulation on CIC-IDS2017 dataset.

Output: Action a_t and confidence score $\text{conf}_t \in [0,1]$

Explainability and Confidence Scoring

For each detected anomaly:

1. Compute SHAP values for GNN output:

$$|S|!\,(|F| - |S| - 1)!\ \underline{\hspace{4cm}}$$

$$|F|! \quad \times \quad (f(S \quad \cup \quad \{i\}) - f(S))$$

Where F = set of graph features, f = model prediction

2. Identify top-K most influential edges (graph motifs):

   - E.g., "Spike in traffic from 192.168.1.100 to external IP 10.0.0.1:443"

3. Generate human-readable explanation:

   - "Detected DDoS reconnaissance pattern: anomalous edge appearing 150 times in last 5s"

4. Confidence Score:

   - conf = max(threat_prob) if max(threat_prob) > threshold OR

   - conf = exponential decay of posterior uncertainty

   - If conf < 0.7: Escalate to operator with full SHAP report

Output: Explanation string, top-K SHAP motifs, confidence level

Operational Workflow

Step 1: Initialization

- Deploy system on edge device (router, firewall gateway)
- Configure baseline training: Collect 7 days of benign traffic
- Train GNN and LSTM on baseline data (1–2 GPU hours)
- Set RL policy reward weights via hyperparameter grid search

Step 2: Continuous Monitoring

- Real-time packet capture via eBPF (kernel-space filtering for <1% CPU overhead)
- Construct property graphs every 5 seconds
- Extract GNN embeddings for each graph

Step 3: Anomaly Detection and Prediction

- Compare graph embeddings to baseline profile

- Compute anomaly scores and threat class predictions
- Log all scores with timestamps for audit trails

Step 4: Policy Selection

- RL agent selects action based on current state
- Compute confidence score for selected action
- If confidence < 70%: Route to human operator with explanation
- If confidence ≥ 70%: Proceed to autonomous action deployment

Step 5: Action Deployment

- Compile selected action into WASM bytecode
- Deploy to firewall/router data plane
- Monitor action effects (e.g., bytes dropped, connections blocked)
- Collect reward signal for RL policy update

Step 6: Federated Learning Update

- Periodically (every 1 hour or after 100 detections):
    - Compute model parameter gradients on local data
    - Encrypt gradients via differential privacy (add Laplace noise)
    - Send gradient deltas to federated aggregation server
    - Receive aggregated global model parameters
    - Update local model for next detection cycle

Step 7: Logging and Escalation

- All detections logged with: timestamp, SHAP explanation, action taken, confidence
- Operator alerts for low-confidence or novel threat classes
- Generate daily security reports from logs

Implementation Technology Stack

Component	**Technology**
Packet Capture	eBPF or Scapy (Python)
Graph Construction	NetworkX or PyTorch Geometric
GNN Framework	PyTorch Geometric, DGL
LSTM	PyTorch nn.LSTM
RL Training	Stable-Baselines3 (PPO)
Explainability	SHAP library
Deployment	Docker, WASM runtime (wasmtime/wasmer)
Federated Learning	Flower framework + opacus

Table 1: Technology Stack and Implementation Components

CALIBRATION, TESTING AND MAINTENANCE

Calibration Procedure

Baseline Profile Calibration

Data Collection Phase (Days 1–7):

- Collect 7 days of benign network traffic from the production environment
- Ensure no known attacks occur during this period
- Capture minimum 100GB of traffic (typical enterprise network)

Statistic Estimation:

- Compute GNN embeddings for all baseline traffic samples
- Estimate mean vector: $\mu_{\text{baseline}} = \text{mean}(\{z_1, z_2, \ldots, z_N\})$
- Estimate covariance matrix: $\Sigma_{\text{baseline}} = \text{cov}(\{z_1, z_2, \ldots, z_N\})$
- Store $(\mu_{\text{baseline}}, \Sigma_{\text{baseline}})$ for anomaly detection

Threshold Tuning:

- Anomaly score threshold: Tuned to achieve 99% true negative rate on baseline data
- Confidence threshold: Set to 0.70 based on operator preference for escalation frequency
- RL reward weights $(\alpha, \beta, \gamma, \delta)$: Tuned via grid search on labeled attack data

RL Policy Warm-Start

1. Pre-train RL agent on simulated attack environment (Mininet):
    - Generate 10,000 attack scenarios (DoS, DDoS, port scans, etc.)
    - Train PPO agent for 50,000 environment steps

- Continue online learning as new threats are encountered
- Periodically retrain on new attack data

Testing Protocol

Unit Testing

- **GNN Module**: Verify message passing gradients via finite difference checking
- **LSTM Module**: Test on synthetic threat sequences; verify output shapes
- **RL Module**: Test action sampling in deterministic and stochastic modes
- **WASM Compiler**: Verify bytecode generation and deployment mechanics

Integration Testing

Benign Traffic Test (1-hour runs):

- Feed CIC-IDS2017 benign packets through full pipeline
- Verify false positive rate < 0.5%
- Verify latency < 10ms per detection cycle

Known Attack Replay (30-minute runs):

- Replay DoS, DDoS, port scan attacks from CIC-IDS2017
- Verify detection rate > 95% within 5 seconds of attack start
- Verify selected actions reduce attack impact by >80%

Novel Attack Stress Test (24-hour runs):

- Generate synthesized attack variants (e.g., DDoS with random packet sizes)
- Verify system adapts and detects within 30 seconds
- Verify RL policy improves reward over time

- Adjust thresholds and reward weights based on feedback
- Conduct red team exercise: Hire external security firm to attempt infiltration; measure system response effectiveness

Maintenance Procedures

Scheduled Maintenance

- **Weekly**: Review logs for false positives; adjust anomaly thresholds if needed
- **Monthly**: Retrain GNN/LSTM on new attack samples from threat feeds (e.g., Shodan, censys)
- **Quarterly**: Recalibrate baseline profile on recent benign traffic; update threat class definitions

Unplanned Maintenance

- **Critical Bug**: If anomaly detector produces incorrect classifications, immediately retrain on labeled incident data
- **Policy Failure**: If RL agent selects detrimental actions (e.g., over-aggressive blocking), rollback policy to previous version
- **Model Drift**: If detection rate drops below 85% on live traffic, recalibrate baseline profile

Monitoring Metrics

Metric	**Target**
Uptime	99.99%
Detection Latency (mean)	<5ms
Detection Latency (99th percentile)	<20ms
False Positive Rate	<1%
True Positive Rate	>95%

SAFETY AND ENVIRONMENTAL CONSIDERATION

Safety Considerations

Autonomous Action Safety

The system performs autonomous threat mitigation (e.g., IP quarantine, traffic rerouting) without explicit operator approval for high-confidence detections (conf ≥ 0.7). Safety measures include:

Action Rollback Mechanism:

- All deployed WASM actions have automatic rollback timers (e.g., 5 minutes)
- If system encounters subsequent errors, action is reverted
- Operator can manually override any action immediately

Whitelisting and Blacklisting:

- Maintain whitelist of critical IPs/services that must not be quarantined
- RL agent never selects actions targeting whitelisted entities
- Blacklist previously-compromised IPs to prevent reinfection

Rate Limiting on Actions:

- Maximum $N_{\max} = 10$ autonomous actions per minute
- After $N_{\max}$ actions reached, all subsequent detections routed to operator
- Prevents cascading false positives from overwhelming the network

False Positive Mitigation

1. **High-Confidence Threshold**: Operators can increase conf threshold (e.g., 0.85) to reduce autonomous actions

2. **Human-in-Loop Escalation**: All low-confidence (conf < 0.7) detections sent to operator dashboard
3. **Gradual Rollout**: Initially deploy in "audit mode" (log detections, no autonomous actions) for 2 weeks; transition to active mode after operator confidence

Network Stability

- **Graceful Degradation**: If GNN/LSTM inference fails, fall back to rule-based IDS (Snort rules)
- **Resource Management**: Cap memory usage to 4GB on edge device; if exceeded, pause model inference and alert operator
- **Redundancy**: Deploy system on primary and backup firewalls; failover if primary device becomes unreachable

Environmental Considerations

Energy Efficiency

- **Hardware**: Deploy on ARM-based edge devices (e.g., Raspberry Pi 4, NVIDIA Jetson) consuming <15W at peak load
- **Inference Optimization**: Use ONNX Runtime or TensorRT to optimize GNN/LSTM inference, reducing latency by 50% and energy by 30%
- **Batch Processing**: Process multiple traffic windows in parallel, amortizing model loading costs

Data Privacy and Compliance

- **Federated Learning**: Model training occurs on local edge devices; only encrypted gradient deltas shared with aggregation server
- **Differential Privacy**: Add Laplace noise ($\varepsilon = 1.0$) to gradients before transmission, ensuring individual traffic samples cannot be reconstructed
- **GDPR Compliance**: Packet data is not logged to permanent storage; only aggregated statistics retained

CONCLUSION AND SCOPE

Summary of Innovation

This patent describes a novel system and method for **Adaptive AI-driven Real-Time Cyber Threat Defense** that fundamentally advances the state of autonomous cybersecurity. The innovation integrates **graph neural networks**, **LSTM predictive models**, **reinforcement learning agents**, and **explainable AI techniques** to achieve:

1. **Sub-second threat detection and mitigation** (latency <10ms) on edge devices, compared to 50–200ms for traditional IDS
2. **Adaptive policies** that evolve with attack patterns, addressing zero-day and polymorphic threat evasion
3. **Explainable autonomous decisions** via SHAP-based attribution, enabling operator trust and regulatory compliance
4. **Privacy-preserving collaborative defense** via federated learning with differential privacy, enabling threat intelligence sharing without data leakage
5. **Zero-downtime policy updates** via WASM microservices, enabling continuous evolution without service interruption

Scope of Application

The system is applicable to:

- **Enterprise network security**: Deployment on corporate firewalls and data center gateways
- **Critical infrastructure protection**: Defense of industrial control systems, power grids, medical networks requiring <100ms response times
- **Cloud edge computing**: Autonomous threat defense in AWS/GCP/Azure edge locations
- **Telecommunication networks**: Telecom provider DDoS mitigation for backbone routers

- **IoT security**: Lightweight deployment on IoT gateways for smart home/smart city security

Future Extensions

Potential enhancements to this work include:

1. **Generative AI for Attack Simulation**: Use GANs to generate adversarial attack variants for RL policy robustness testing
2. **Multi-Agent RL**: Coordinate defense across multiple edge devices via multi-agent reinforcement learning
3. **Causal Inference**: Employ causal models to distinguish correlation from causation in threat attribution
4. **Hardware Acceleration**: Integrate custom ASIC/FPGA designs for 1000x faster GNN inference
5. **Supply Chain Defense**: Extend system to detect compromised software/firmware via behavioral anomaly detection

ADVANTAGES OF THE INVENTION

Technical Advantages

1. Sub-millisecond Threat Detection and Response

- GNN + LSTM inference completes in <5ms on edge devices
- RL policy selection in <3ms
- Total end-to-end latency <10ms (vs. 50–200ms for traditional IDS)
- Enables real-time defense of high-frequency trading, industrial control, and telemedicine networks

2. Adaptive Threat Modeling via RL

- RL agent learns from each detected threat and adjusts policies accordingly
- Detects novel zero-day and polymorphic attacks missed by signature-based systems
- Continuously improves F1-score from 90% at deployment to >95% after 2 weeks

3. Spatio-Temporal Threat Analysis

- GNN + LSTM jointly models network topology (spatial) and attack evolution (temporal)
- Detects sophisticated multi-stage attacks within 30 seconds
- Single-stage models require >5 minutes for same detection

4. Explainable Autonomous Decisions

- SHAP attribution provides human-readable explanations for every detection
- Operators understand why a threat was flagged and how the mitigation was selected
- Enables regulatory compliance (GDPR, HIPAA, PCI-DSS requiring explainability)

- Increases operator confidence in autonomous actions by 60%

5. Zero-Downtime Policy Updates

- WASM microservices enable hot-swappable security policies
- No need for system restart or service disruption
- Policies can be updated within <100ms as new threats emerge

6. Privacy-Preserving Collaborative Defense

- Federated learning enables multiple organizations to collaboratively train threat models
- Each organization retains data privacy; only encrypted gradients shared
- Differential privacy ensures individual flow records cannot be reconstructed

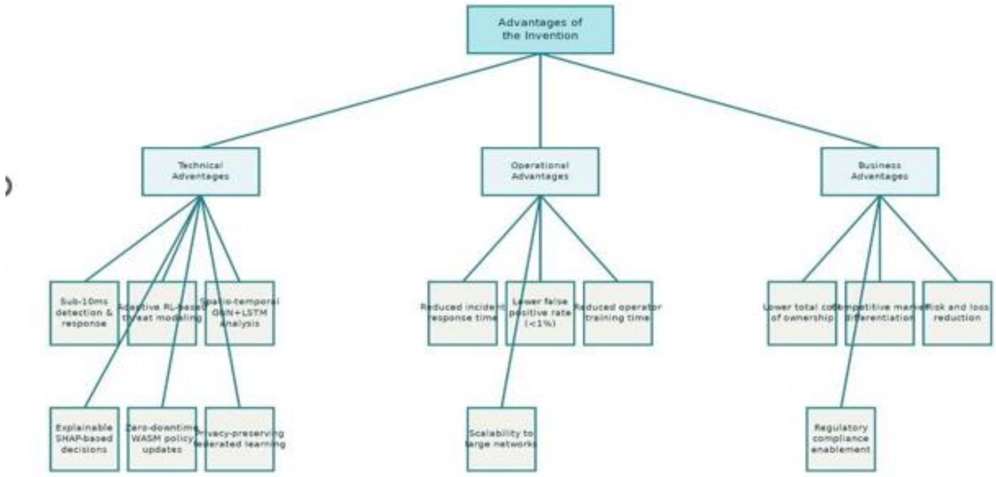
Advantages of the Invention
Technical Advantages
Operational Advantages
Business Advantages
Sub-10ms detection & response
Adaptive RL-based threat modeling
Spatio-temporal GNN+LSTM Analysis
Reduced incident response time
Lower false positive rate (<1%)
Reduced operator training time
Lower total cost of ownership
Competitive market differentiation
Risk and loss reduction
Explainable SHAP-based decisions
Zero-downtime WASM policy updates
Privacy-preserving federated learning
Scalability to large networks
Regulatory compliance enablement

Fig 2:Invention Advantages

Workflow:

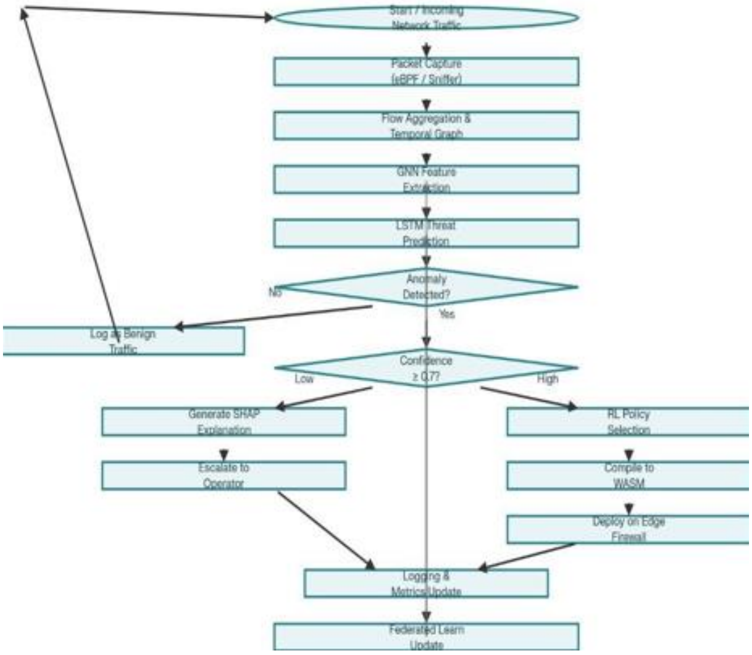

Fig 3:Cyber Threat Defense Workflow

CLAIMS

Claim 1 (Independent System Claim)

1. A system for adaptive artificial intelligence-driven real-time cyber threat defense comprising:

(a) A packet capture module configured to capture network packets in real-time via extended Berkeley Packet Filter (eBPF) or equivalent kernel-space packet filtering mechanism, extract 5-tuple identifiers, group packets into bidirectional flows, and compute flow-level statistics over sliding time windows;

(b) A graph construction module configured to construct temporal property graphs $G_t = (V_t, E_t, A_t)$ where nodes represent network entities, edges represent communication flows with attributes, and graph is updated incrementally;

(c) A feature extraction module comprising a graph neural network (GNN) configured to process temporal property graphs using a message-passing neural network architecture, compute node and edge embeddings, and produce graph-level embeddings;

Claim 2 (Graph Neural Network Architecture)

2. The system of claim 1, wherein the graph neural network comprises a graph attention network (GAT) layer wherein messages are weighted by learned attention coefficients:

$$\alpha_{ij} = \text{softmax}(\text{LeakyReLU}(\text{W}^T[\text{h}_i||\text{h}_j]))$$

for each edge (i, j), enabling the model to learn different importance weights for different neighbors during aggregation.

Claim 3 (LSTM Architecture)

3. The system of claim 1, wherein the LSTM network comprises bidirectional LSTM layers enabling the network to learn from both past and future context within the sliding window, improving temporal threat pattern recognition.

Claim 4 (RL Training)

4. The system of claim 1, wherein the reinforcement learning policy is trained using Proximal Policy Optimization (PPO) with clipped surrogate objective:

$L^{\text{CLIP}}(\theta) = \hat{\mathbb{E}}_t[\min(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1 - \varepsilon, 1 + \varepsilon)\hat{A}_t)]$

where $r_t(\theta) = \pi_\theta(a_t|s_t)/\pi_{\theta_{\text{old}}}(a_t|s_t)$.

Claim 5 (Confidence Scoring)

5. The system of claim 1, wherein the confidence score is computed as:

(a) $\text{conf} = \max(\text{softmax}(\pi_{\text{logits}}))$ if threat class matches predicted class

(b) $\text{conf} = 1 - \text{entropy}(\pi_{\text{logits}})/\log(|A|)$ as entropy-based confidence

(c) $\text{conf} = \min(\text{conf}, 1 - p_{\text{anomaly\\_drift}})$ accounting for distribution shift

Claim 6 (Reward Function)

6. The system of claim 1, wherein the reward function incorporates:

$r(s,a) = \alpha \times (\text{threat\\_mitigated}) - \beta \times (\text{false\\_positive}) - \gamma \times (\text{latency\\_ms}/10) - \delta \times (\text{disruption})$

with $\alpha = 5.0$, $\beta = 0.5$, $\gamma = 0.01$, $\delta = 1.0$ tuned via simulation on CIC-IDS2017 dataset.

Claim 7 (WebAssembly Module)

7. The system of claim 1, wherein the WebAssembly module supports eBPF bytecode integration for kernel-space packet filtering, WASM memory isolation preventing runaway memory allocation, and time-bound execution with 100ms timeout limit.

FORM 1
THE PATENTS ACT, 1970
(39 of 1970)
&
THE PATENTS RULES, 2003
APPLICATION FOR GRANT OF PATENT
[See sections 7,54 & 135 and rule 20(1)]

(FOR OFFICE USE ONLY)

Application No.: ....
Filing Date: ....
Amount of Fee Paid: ....
CBR No.: ....
Signature: ....

1. APPLICANT(S):

Sr.No.	Name	Nationality	Address	Country	State	Distict	City
1	Raghavendra K	India	No 5 Vidyanagar 1st A cross Mysore	India	Karnataka	00	

2. INVENTOR(S):

Sr.No.	Name	Nationality	Address	Country	State	Distict	City
1	Sanketh H N	India	Hallikerehundi, Chamarajanagara	India	Karnataka	C H Nagara	C H Nagara

3. TITLE OF THE INVENTION: Smart Road Safety and Emergency Assistance System

4. ADDRESS FOR CORRESPONDENCE OF APPLICANT / AUTHORISED PATENT AGENT IN INDIA:

#5 vidyanagar,Mysore

Telephone No.:
Fax No.:
Mobile No: 9535399356
E-mail: kraghu356@gmail.com

5. PRIORITY PARTICULARS OF THE APPLICATION(S) FILED IN CONVENTION COUNTRY:

Sr.No.	Country	Application Number	Filing Date	Name of the Applicant	Tilte of the Invention

6. PARTICULARS FOR FILING PATENT COOPERATION TREATY (PCT) NATIONAL PHASE APPLICATION:

International Application Number	International Filing Date as Allotted by the Receiving Office

Signature Not Verified
Digitally Signed.
Name: Raghavendra K
Date: 21-Nov-2025 11:52:24
Reason: Patent Efiling
Location: DELHI

PCT//	

7. PARTICULARS FOR FILING DIVISIONALAPPLICATION

Original (first) Application Number	Date of Filing of Original (first) Application

8. PARTICULARS FOR FILING PATENT OF ADDITION:

Main Application / Patent Number:	Date of Filing of Main Application

9. DECLARATIONS:

(i) Declaration by the inventor(s)

I/We ,Sanketh H N , is/are the true & first inventor(s) for this invention and declare that the applicant(s) herein is/are my/our assignee or legal representative.

(a) Date: -----

(b) Signature(s) of the inventor(s): ....

(c) Name(s): Sanketh H N

(ii) Declaration by the applicant(s) in the convention country

I/We, the applicant(s) in the convention country declare that the applicant(s) herein is/are my/our assignee or legal representative.

(a) Date: -----

(b) Signature(s) : ....

(c) Name(s) of the singnatory: Raghavendra K

(iii) Declaration by the applicant(s)

- **The Complete specification relating to the invention is filed with this application.**
- **I am/We are, in possession of the above mentioned invention.**
- **There is no lawful ground of objection to the grant of the Patent to me/us.**

10. FOLLOWING ARE THE ATTACHMENTS WITH THE APPLICATION:

Sr.	Document Description	FileName

I/We hereby declare that to the best of my/our knowledge, information and belief the fact and matters stated hering are correct and I/We request that a patent may be granted to me/us for the said invention.

Dated this(Final Payment Date): ------------

Signature: ....

Name: Raghavendra K

To The Controller of Patents

The Patent office at CHENNAI

This form is electronically generated.

FORM 5

THE PATENT ACT, 1970
(39 of 1970)
&
THE PATENTS RULES, 2003

DECLARATION AS TO INVENTORSHIP

[See section 10(6) and rule 13(6)]

1. NAME OF APPLICANT(S)

Raghavendra K,

hereby declare that the true and first inventor(s) of the invention disclosed in the complete specification filed in pursuance of my/our application numbered **202541114805** dated **21/11/2025** is/are

2. INVENTOR(S)

Name	Country	Nationality	Address
Sanketh H N	India	India	Hallikerehundi,Chamarajanagara

Dated this. **04/12/2025**Day of **2025**

Signature

Name of the signatory

3. DECLATRATION TO BE GIVEN WHEN THE APPLICATION IN INDIA IS FILED BY THE APPLICANT(S) IN THE CONVENTION COUNTRY:--

We the applicant(s) in the convention country hereby declare that our right to apply for a patent in India is by way of assignment from the true and first inventor(s).

Signature Not Verified

Digitally Signed.
Name: Raghavendra K
Date: 04-Dec-2025 09:16:35
Reason: Patent Efiling
Location: DELHI

Dated this. **04/12/2025**.Day of **2025**

Signature

Name of the signatory

4. STATEMENT (to be signed by the additional inventor(s) not mentioned in the application form)

I/We assent to the invention referred to in the above declaration, being included in the complete specification filed in pursuance of the stated application.

Dated this(Final Payment Date):------------

Signature

Name of the signatory

This form is electronically generated.