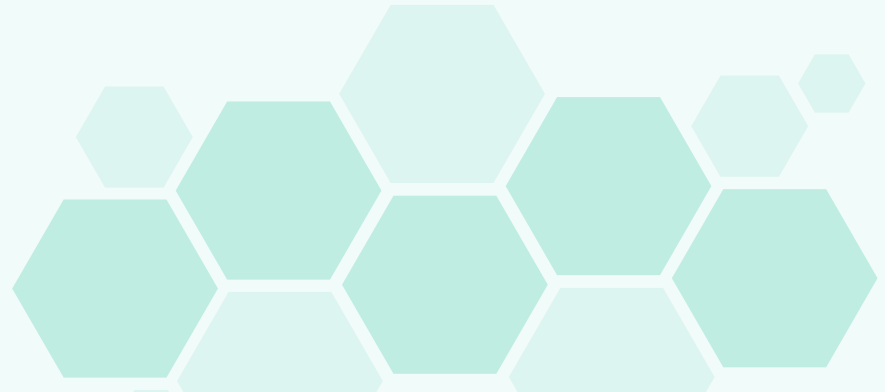


Securing Dynamic Robotic Behavior in Unpredicted Environments: Enhancing Trust through Adaptive Learning and Cyber Defense

Giselle Roman | Dr. Yugyung Lee | 06/23/2025



Problem Statement Overview

In unpredictable environments like post-disaster zones, robots must make quick, secure decisions. Cyber anomalies can disrupt trust and coordination, so there's a need for adaptive, anomaly-aware task assignment systems that respond in real time.

Hypotheses

1. GNNs will outperform traditional methods (e.g., autoencoders, statistical baselines) in detecting adversarial inputs and irregular robot trajectories.
2. Using GNNs in robot controllers will boost adaptability and response time in multi-agent systems when anomalies occur.

Research Questions

1. Can GNNs reliably detect real-time anomalies in robots operating in unpredictable environments?
2. How does GNN-based anomaly detection impact trust and recovery time in simulated robotic systems?

Paper 1: GNN-Based Anomaly Detection

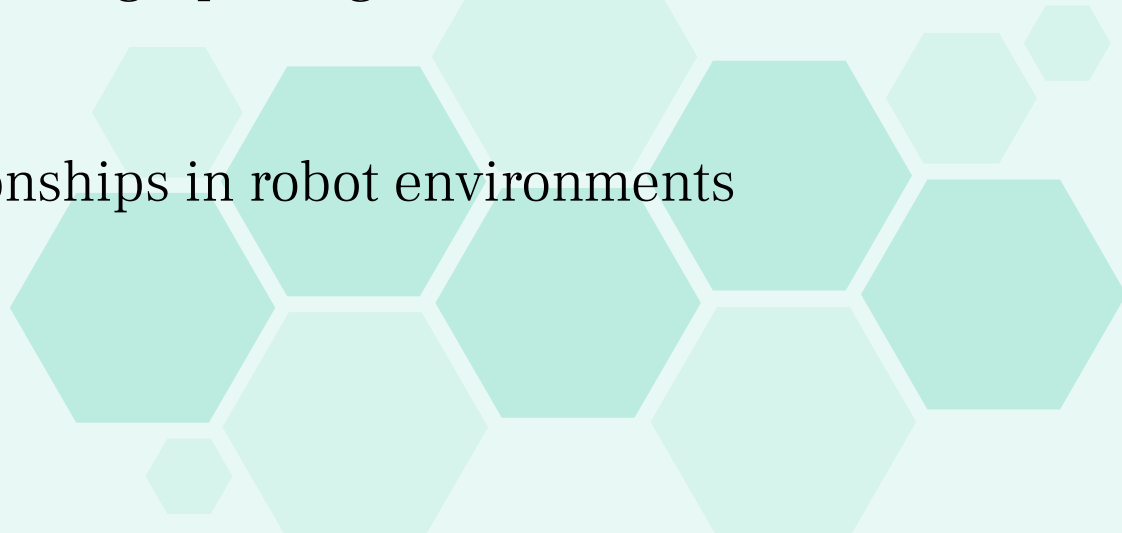
- Survey on using GNNs for detecting anomalies in graphs
- Anomalies in graphs
 - Node: outliers based on attributes/structure
 - Edge: unexpected or unusual connections
 - Subgraph & Graph-level: structural or community deviations
- GNN techniques
 - Graph Autoencoders (GAE): unsupervised learning using reconstruction errors
 - Dynamic Graph Convolutional Network (DGCN): track changes in graphs for real-time detection

Paper 2: Deep Reinforcement Learning with GNNs

- DRL + GNN integration
 - Combines decision making ability of DRL with structure-aware reasoning of GNNs
- Adversarial Learning
 - Deep Q-learning used to poison graphs (inject fake nodes/edges)
- Combinatorial Optimizations
 - Finding the best solution from a set of possibilities
- Partially Observable Markov Decision Process (POMDP)
 - Mapping from past observations

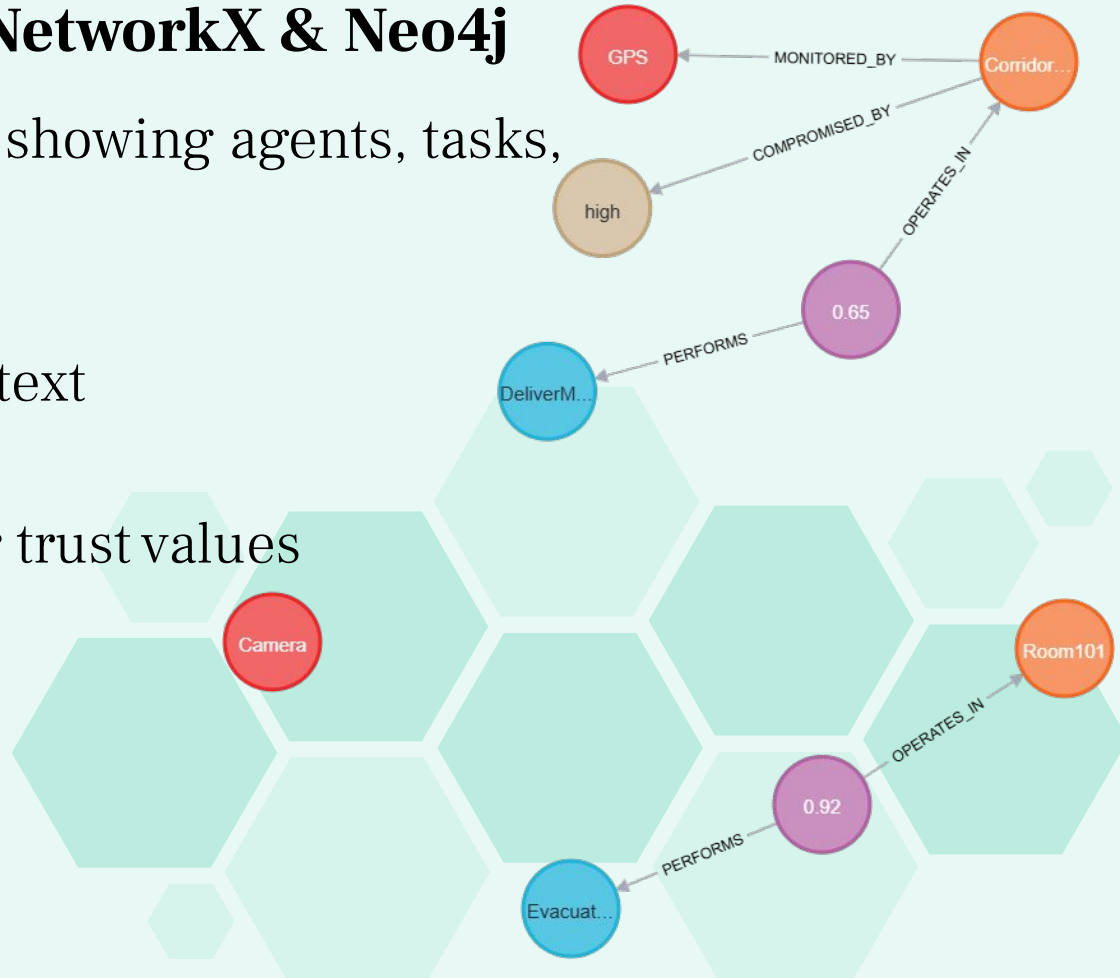
Graph Modeling with NetworkX & Neo4j

- Knowledge graph to simulate trust, task execution, and location context in disaster scenarios
- NetworkX
 - Generated synthetic graphs (agents, tasks, sensors, risks)
- Neo4j
 - Visualize relationships in robot environments



Graph Modeling with NetworkX & Neo4j

- Constructed a graph showing agents, tasks, and locations
 - Room101
- Added edges for context
 - PERFORMS
- Assigned weights for trust values
 - 0.92



Graph Modeling with NetworkX & Neo4j

- Foundation for anomaly detection using GNNs
- Allows robotic agents to reason about
 - Tasks
 - Threats
 - Space



Reinforcement Learning with Gymnasium

- Train a robot agent to navigate a simulated environment
 - Optimizing task performance
- Gymnasium
 - Simulated environments and training RL agents
- Q-Learning
 - Tabular RL method used as baseline



Reinforcement Learning with Gymnasium

- Defined a simple environment
 - 16 states and 4 actions
- Trained a Q-learning agent over 100 episodes
- Q-table showing action-value estimates
- Achieved 62% success rate in task completion

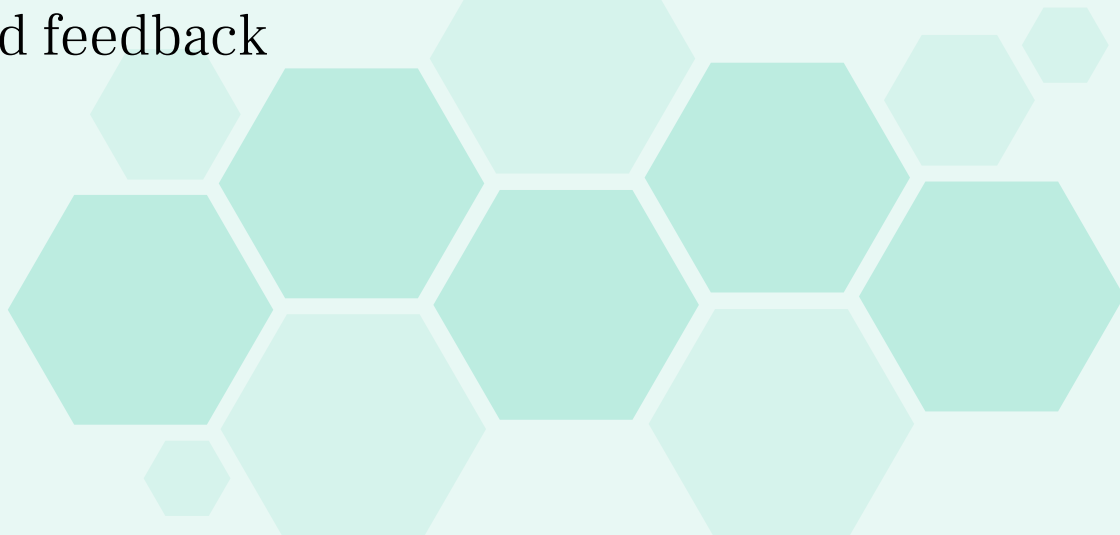
Trained Q-table:

```
[[1.03908686e-01 8.32207428e-03 7.07515848e-03 6.23802887e-03]
 [7.79828672e-04 5.16035425e-05 2.84288668e-04 1.03228989e-01]
 [7.40731367e-03 3.90385060e-03 3.11017979e-03 1.63147112e-01]
 [8.50533791e-05 3.21756543e-03 7.36179746e-04 4.94965380e-02]
 [5.93158936e-02 2.04596596e-03 1.36843134e-03 1.43969077e-03]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [3.97143285e-02 6.07940025e-05 3.44989345e-04 1.80151224e-06]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [6.38971942e-05 2.78116848e-04 1.16293475e-03 9.93615767e-02]
 [0.00000000e+00 5.93159628e-01 1.89483641e-03 8.52681546e-04]
 [3.23464396e-01 2.67480494e-04 5.99821041e-04 3.30745283e-04]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [1.13247624e-03 0.00000000e+00 5.90874070e-01 5.39867832e-03]
 [0.00000000e+00 0.00000000e+00 9.24186898e-01 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]]
```

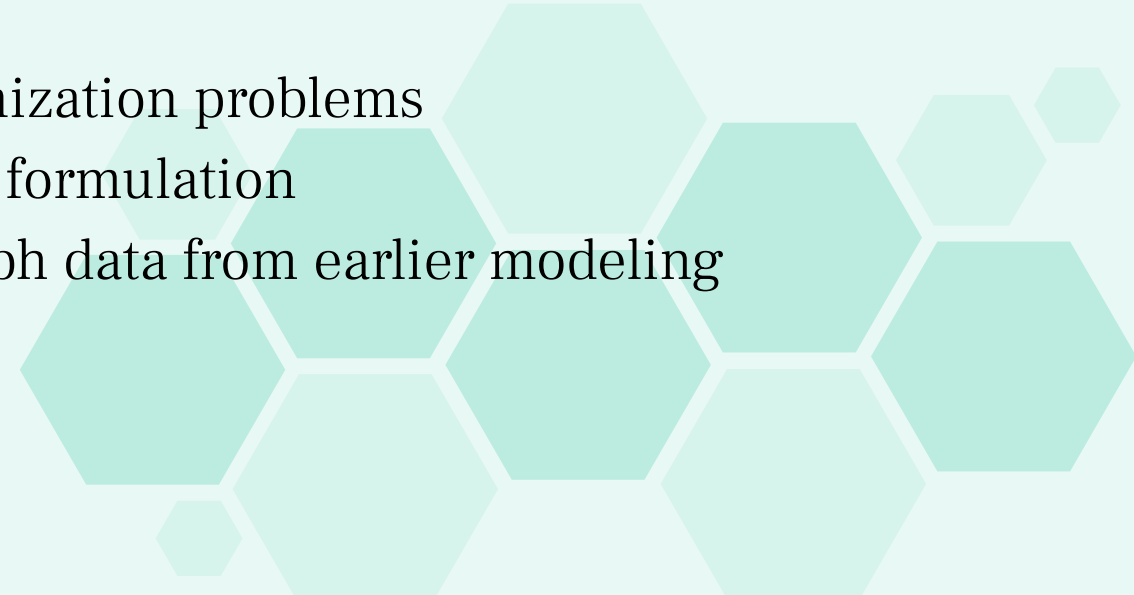
Success rate over 100 episodes: 62%

Reinforcement Learning with Gymnasium

- Baseline for comparing advanced models
- Validated how well robots adapt and learn optimal behavior with limited feedback



Quantum Optimization with QUBO

- Solve a task assignment problem
 - Agents are matched to tasks based on trust and priority
 - Qiskit + DQcplex
 - Modeling optimization problems
 - Generated a QUBO formulation
 - Used synthetic graph data from earlier modeling
- 

Quantum Optimization with QUBO

- QUBO model for 3 agents and 3 tasks
- Maximize trust scores
 - One task per agent constraint
- Encoded constraints with slack variables
- File with binary variables and cost coefficients

```
QUBO created:
\ This file has been generated by D0cplex
\ ENCODING=ISO-8859-1
\ Problem name: task_assignment

Minimize
obj: - 17 x_agent_0_task_0 - 17 x_agent_1_task_1 - 17 x_agent_2_task_2
      - 8 c0@int_slack@0 - 8 c1@int_slack@0 - 8 c2@int_slack@0
      - 8 c3@int_slack@0 - 8 c4@int_slack@0 - 8 c5@int_slack@0 + [
      16 x_agent_0_task_0^2 + 16 x_agent_0_task_0*c0@int_slack@0
      + 16 x_agent_0_task_0*c3@int_slack@0 + 16 x_agent_1_task_1^2
      + 16 x_agent_1_task_1*c1@int_slack@0 + 16 x_agent_1_task_1*c4@int_slack@0
      + 16 x_agent_2_task_2^2 + 16 x_agent_2_task_2*c2@int_slack@0
      + 16 x_agent_2_task_2*c5@int_slack@0 + 8 c0@int_slack@0^2
      + 8 c1@int_slack@0^2 + 8 c2@int_slack@0^2 + 8 c3@int_slack@0^2
      + 8 c4@int_slack@0^2 + 8 c5@int_slack@0^2 ]/2 + 24

Subject To

Bounds
0 <= x_agent_0_task_0 <= 1
0 <= x_agent_0_task_1 <= 1
0 <= x_agent_0_task_2 <= 1
0 <= x_agent_1_task_0 <= 1
0 <= x_agent_1_task_1 <= 1
0 <= x_agent_1_task_2 <= 1
0 <= x_agent_2_task_0 <= 1
0 <= x_agent_2_task_1 <= 1
0 <= x_agent_2_task_2 <= 1
0 <= c0@int_slack@0 <= 1
0 <= c1@int_slack@0 <= 1
0 <= c2@int_slack@0 <= 1
0 <= c3@int_slack@0 <= 1
0 <= c4@int_slack@0 <= 1
0 <= c5@int_slack@0 <= 1

Binaries
x_agent_0_task_0 x_agent_0_task_1 x_agent_0_task_2 x_agent_1_task_0
x_agent_1_task_1 x_agent_1_task_2 x_agent_2_task_0 x_agent_2_task_1
x_agent_2_task_2 c0@int_slack@0 c1@int_slack@0 c2@int_slack@0 c3@int_slack@0
c4@int_slack@0 c5@int_slack@0

End
```

Quantum Optimization with QUBO

- Modeled real world robotic decisions
 - Dynamic task reassignment under constraints
- Crucial in disaster response or adversarial scenarios



Next Steps

Week	Focus
Week 4 (June 17–21)	<ul style="list-style-type: none">- Explored trust-based task allocation- Created QUBO model with Qiskit- Ran RL simulations in Gymnasium- Analyzed graph anomaly papers
Week 5 (June 24–28)	<ul style="list-style-type: none">- Implement GNN anomaly detection baseline- Refine simulation setup- Add initial trust metrics to graphs
Week 6 (July 1–5)	<ul style="list-style-type: none">- Integrate GNN outputs with trust feedback- Begin edge-case/adversarial testing
Week 7 (July 8–12)	<ul style="list-style-type: none">- Finalize anomaly detection experiments- Run full trust-aware coordination tests
Week 8 (July 15–19)	<ul style="list-style-type: none">- Revise report/slides- Organize GitHub
Week 9 (July 22–26)	<ul style="list-style-type: none">- Practice presentation- Final cleanup
Final Week (July 29–31)	<ul style="list-style-type: none">- Give final presentation

References

- Kim, Hwan, Byung Suk Lee, Won-yong Shin, and Sungsu Lim. “Graph Anomaly Detection with Graph Neural Networks: Current Status and Challenges.” IEEE Access, vol. 10, 2022, pp. 111820–29, <https://doi.org/10.1109/access.2022.3211306>.
- Munikoti, Sai, Deepesh Agarwal, Laya Das, Mahantesh Halappanavar, and Balasubramaniam Natarajan. “Challenges and Opportunities in Deep Reinforcement Learning with Graph Neural Networks: A Comprehensive Review of Algorithms and Applications.” IEEE Transactions on Neural Networks and Learning Systems, Institute of Electrical and Electronics Engineers, Jan. 2023, pp. 1–21, <https://doi.org/10.1109/tnnls.2023.3283523>.

