

AI-Native Routing and Network Layer Intelligence

Lecture 6

By Dr. Tarfa Yaseen Hamed

Apr. 2026

Outline

- ▶ Introduction: Beyond Static Routing
- ▶ The Evolution of Layer 3
- ▶ The 2026 Problem: The "Static Rule" Crisis
- ▶ The AI Pivot: Intent-Based Networking (IBN)
- ▶ Key AI Pillars for the Network Layer
- ▶ Case Study 1: Traffic Engineering with LSTMs
- ▶ Case Study 2: Adaptive Routing with DRL
- ▶ Case Study 3: Topology Learning with Graph Neural Networks (GNN)
- ▶ Case Study 4: Intent-Based Slicing (SLA Management)
- ▶ Case Study 5: Graph Neural Networks (GNN) for Topology Sensing
- ▶ Mapping AI Models to Network Layer Tasks

Introduction: Beyond Static Routing

- The Network Layer is responsible for packet forwarding and routing across multiple hops from source to destination.
- Traditional routing protocols (OSPF, BGP) rely on static metrics like "shortest path" or "lowest cost".
- **The Problem:** In 6G/2026, network topology is highly dynamic (e.g., satellite-to-drone-to-ground) and traffic is "Ultra-Dense," making static rules inefficient.
- **Where AI fits in:**
 - **Traffic Engineering:** Predicting congestion before it happens.
 - **Dynamic Routing:** Finding the best path based on real-time latency rather than hop count.
 - **Security:** Detecting routing attacks or "Sinkhole" anomalies using pattern recognition.

The Evolution of Layer 3

- **The Core Responsibility (The "Classic" View)**
- The Network Layer is fundamentally the "post office" of the stack. While the Data Link Layer moves data between two nodes on the same network, the Network Layer manages:
 - **Path Determination:** Choosing the route from source to destination across multiple intermediate hops.
 - **Forwarding:** The actual hardware process of moving a packet from an input interface to the correct output interface.

The 2026 Problem: The "Static Rule" Crisis

- Traditional protocols like OSPF (Open Shortest Path First) or BGP (Border Gateway Protocol) operate on **fixed rules**.
- They typically calculate the "shortest path" based on hop count or a **static cost** assigned to a link.
- **Why this fails today:** In ultra-dense environments, the "shortest" path is **often the** most congested.
- **Dynamic Topology:** With the rise of Non-Terrestrial Networks (NTN) like **drones and** low-orbit satellites, the "map" of the network changes every second. **Static tables** cannot converge fast enough to keep up.

The AI Pivot: Intent-Based Networking (IBN)

- ▶ This is where AI changes the philosophy of the introduction. The move is from "How to route" to "What is the goal?"
 - **Resource Allocation:** AI doesn't just find a path; it decides which traffic gets the premium path based on its importance.
 - **Proactive Congestion Management:** Instead of reacting when a router's buffer is full, AI predicts latency spikes and reroutes traffic before the buffer overflows.

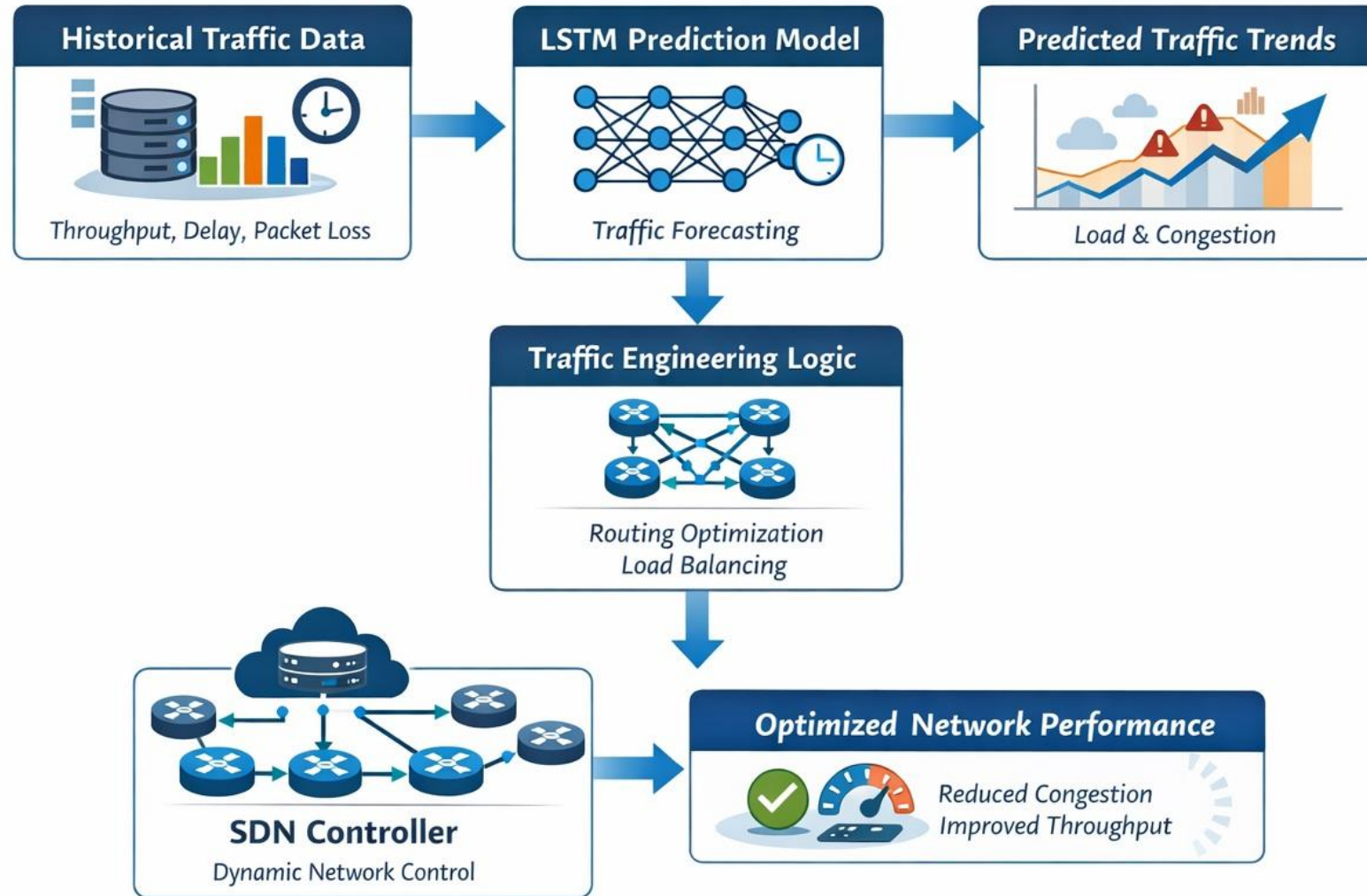
Key AI Pillars for the Network Layer

Pillar	Technical Role	AI Model Typically Used
Traffic Engineering	Predicting peak loads and balancing traffic across the whole mesh.	LSTMs (for time-series prediction).
Adaptive Routing	Finding paths based on real-time "Health" of the link rather than distance.	Deep Reinforcement Learning (DRL) .
Topology Learning	Understanding the complex relationships between thousands of moving nodes.	Graph Neural Networks (GNN) .

Case Study 1: Traffic Engineering with LSTMs

- ▶ **The Scenario:** A "Smart City" network experiencing unpredictable traffic surges due to a major sports event and autonomous vehicle coordination.
 - **The Traditional Failure:** Standard OSPF routing sends all packets along the "shortest path." This creates a bottleneck (hotspot) on the main fiber backbone while neighboring links sit at 10% usage.
 - **The AI Pillar (Predictive Scaling):** An LSTM (Long Short-Term Memory) network monitors historical traffic patterns.
 - **The Solution:** The LSTM predicts a congestion spike 5 minutes before it happens by recognizing the "rhythm" of the city's data flow.
 - **The Outcome:** The network proactively "Load Balances" the traffic, distributing the load across the mesh. This prevents Tail Latency spikes that would otherwise stall autonomous vehicle sensors.

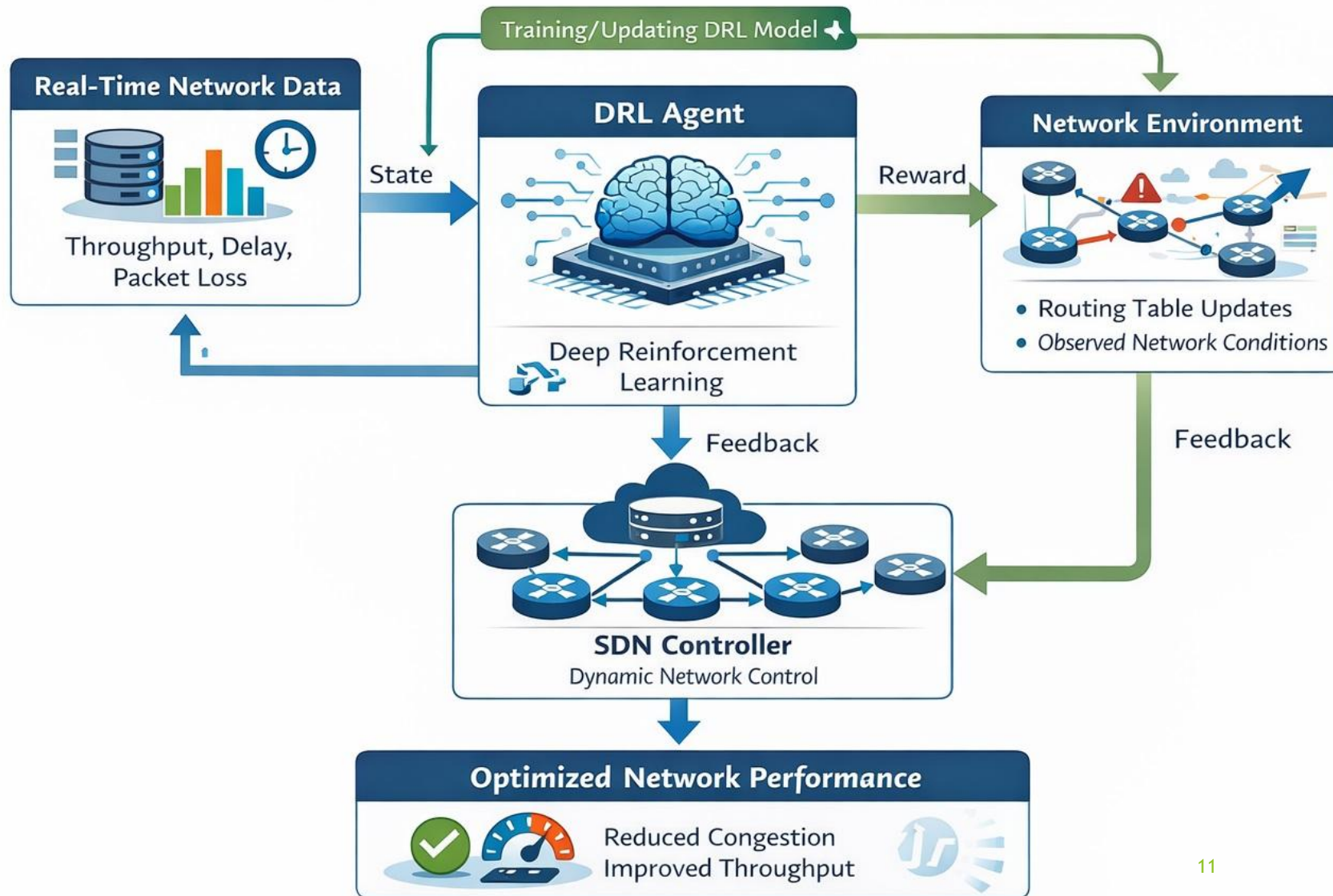
Predictive Traffic Engineering with LSTM



Case Study 2: Adaptive Routing with DRL

- ▶ **The Scenario:** A 6G Non-Terrestrial Network (NTN) where routers are mounted on moving drones and Low-Earth Orbit (LEO) satellites.
- **The Traditional Failure:** Routers use "Fixed Rules" like hop counts. Because the satellites are moving, the routing tables can never "converge" fast enough, leading to massive packet loss.
- **The AI Pillar (Reinforcement Learning):** A Deep Q-Network (DQN) acts as the Routing Agent.
- **The Solution:** The Agent observes the "State" (satellite position, link signal-to-noise ratio, and Doppler Spread). It selects a path (Action) that maximizes the "Reward" (high throughput and low delay).
- **The Outcome:** The AI learns to "hop" packets across moving satellites with near-instant convergence, maintaining a stable connection for remote areas.

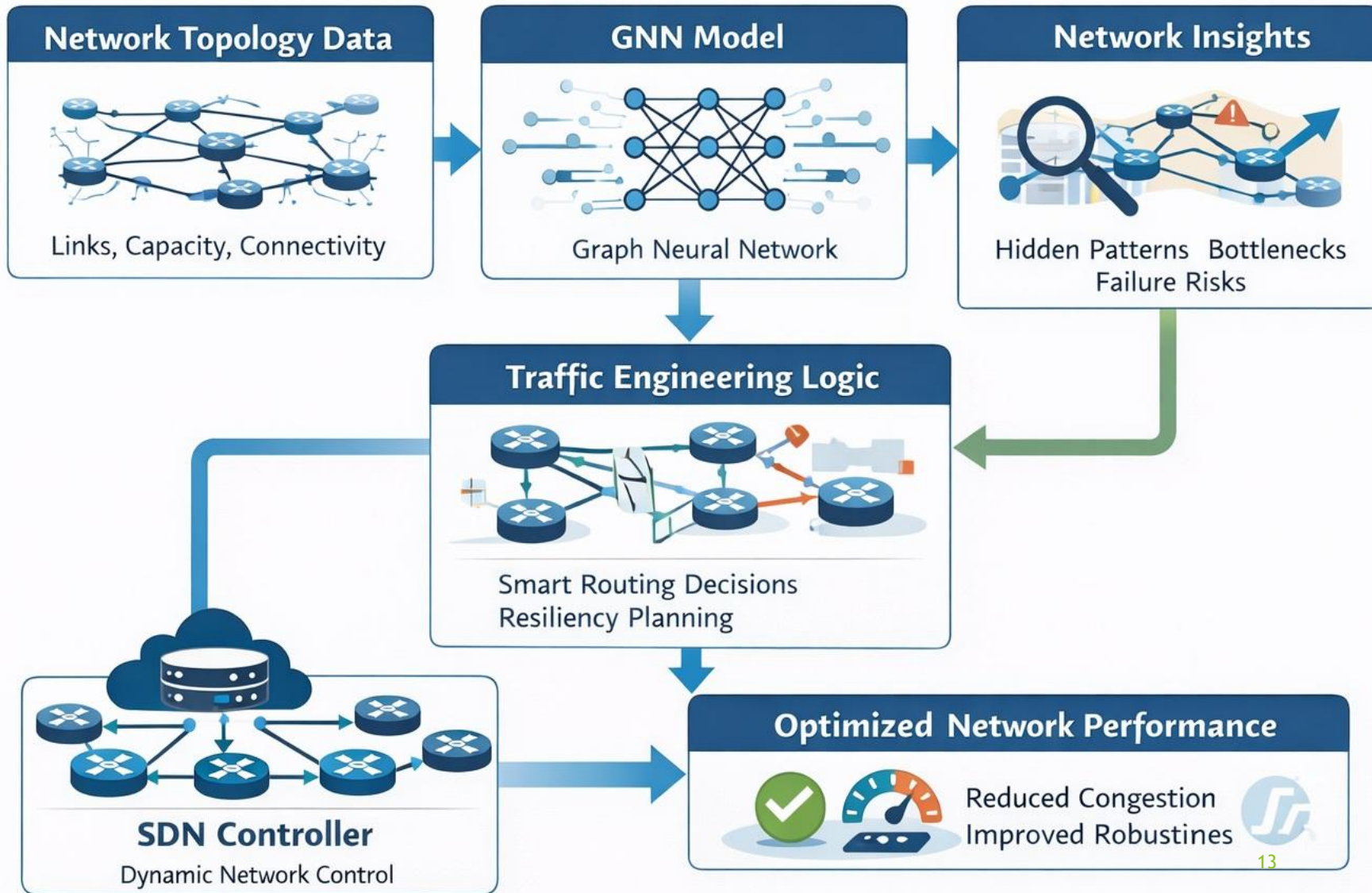
Adaptive Routing with Deep Reinforcement Learning (DRL)



Case Study 3: Topology Learning with Graph Neural Networks (GNN)

- ▶ **The Scenario:** A massive IoT deployment in a factory where 5,000 sensors are constantly joining and leaving the network.
 - **The Traditional Failure:** Traditional routing protocols use "Hello" packets to map the network. In an ultra-dense environment, these "Hello" packets cause a "Collision Storm," using up all the bandwidth.
 - **The AI Pillar (GNN-based Sensing):** The network uses Graph Neural Networks to understand the spatial relationships between nodes.
 - **The Solution:** Unlike traditional AI, GNNs can process the "Graph" structure of the network. If a group of 100 sensors goes offline, the GNN doesn't need to re-calculate everything; it instantly generalizes the new topology.
 - **The Outcome:** The network maintains "Zero Collision" routing even as the physical map of the factory changes every second.

Topology Learning with Graph Neural Networks (GNN)



Case Study 4: Intent-Based Slicing (SLA Management)

- ▶ **The Scenario:** A single physical network backbone must simultaneously support a Remote Robotic Surgery (URLLC) and a Guest Wi-Fi (eMBB).
 - **The Traditional Failure:** "Proportional Fair" scheduling gives everyone a "fair" share. During high traffic, the surgery packet gets delayed by the Guest Wi-Fi traffic, causing an SLA Violation.
 - **The AI Pillar (Cross-Layer Slicing):** A DDPG Agent (Deep Deterministic Policy Gradient) manages the "Network Slice".
 - **The Solution:** The AI receives "Intent" (e.g., "The surgery must have <1ms latency"). It monitors Buffer Occupancy and CSI.
 - **The Outcome:** The AI "slices" the network so effectively that the SLA Violation Rate drops from 5% to <0.1%, ensuring life-critical data always has a clear path.

Case Study 5: Graph Neural Networks (GNN) for Topology Sensing

- ▶ While RNNs is used for time-series channel estimation, the Network Layer uses **GNNs** to understand the "map" of the network.
- ▶ **The Challenge:** Traditional AI struggles when the network changes (e.g., a router goes offline).
- ▶ **The Solution:** GNNs treat the network as a "Graph" (Nodes and Edges).
- ▶ **Implementation:** The AI learns the relationships between nodes. If a link fails, the GNN instantly "generalizes" and predicts the new optimal path without needing to be retrained from scratch.

Mapping AI Models to Network Layer Tasks

Network Layer Task	Primary AI Model	Why this model?	Engineering Goal
Traffic Engineering	RNN / LSTM	It has "Memory" to track time-series data and predict future congestion.	Prevent "Hotspots" and optimize Load Balancing.
Adaptive Routing	DRL (Deep Reinforcement Learning)	It learns through "Trial and Error" to find optimal paths in changing environments.	Near-instant convergence in dynamic topologies (Drones/Satellites).
Topology Mapping	GNN (Graph Neural Networks)	It is designed to understand the "Relationship" (edges) between "Nodes" (routers).	Map ultra-dense IoT networks
Resource Slicing	DDPG / Actor-Critic	It manages continuous resource variables (bandwidth/power) to meet strict intents.	Minimize SLA Violation Rates to <0.1%.
Anomaly Detection	Autoencoders (CNN-based)	It identifies "Normal" patterns and flags deviations (DDoS or Sinkhole attacks).	Secure the control plane against malicious routing updates.