Buffalo Accident Risk Prediction & Resource Allocation

Fagun Patel, Meghna Shekhar, Sahil Sawant (CSE 4/546 Reinforcement Learning under Prof. Alina Vereshchaka)

Introduction

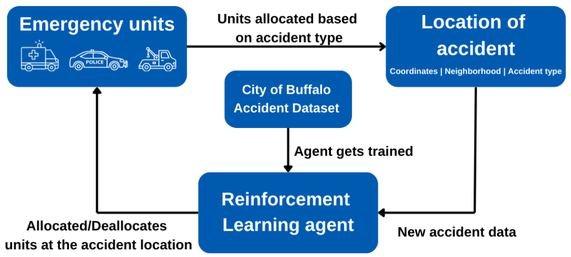
Traffic accidents in urban areas pose a critical challenge, often leading to delayed emergency response and increased casualties. On an average, approximately 10,000 incidents were reported annually. Traditional resource allocation struggles to adapt to rapidly changing risk patterns across a city due to uncertainity.

Our Methodology

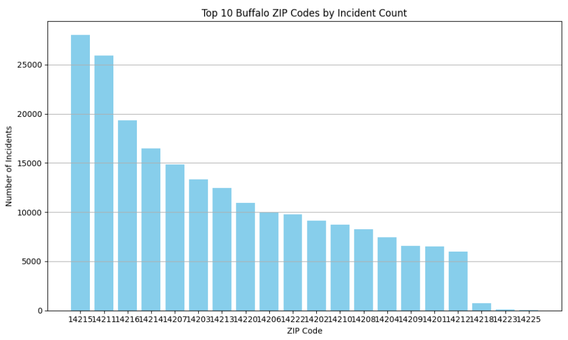
This project presents a Reinforcement Learning (RL) approach that optimizes the deployment of emergency services to minimize response times and improve public safety by predicting high-risk accident zones in real-time .

RL for Emergency Resource Allocation

RL is a branch of machine learning where an agent learns decision-making by interacting with an environment and receiving feedback through rewards or penalties. Our RL model is trained to allocate emergency resources effectively based on accident risk patterns across Buffalo.



Buffalo Traffic Analysis

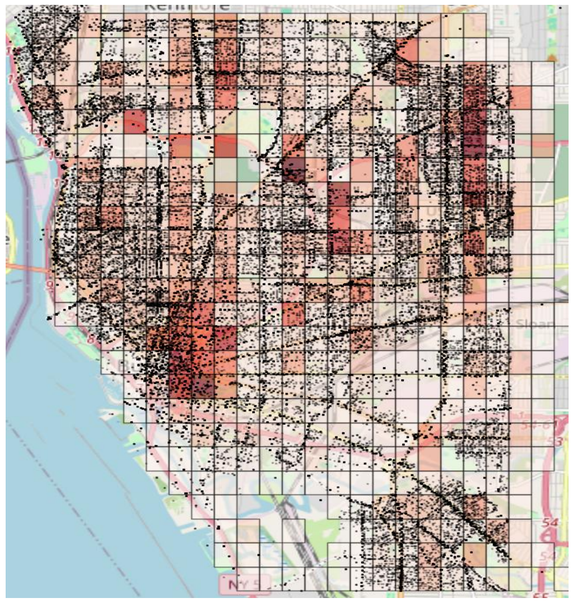


Occurrences of Accidents in a particular zip code

We analyzed Buffalo data by location (latitudes & longitudes), accident type, and neighborhood, with initial visualizations highlighting regions with higher incident frequencies across ZIP codes and police districts.

Environment	
State	[0,1] Predicted accident risk and current resource allocation for each zone.
Action	Allocate or deallocate resources; actions are continuous values scaled into unit movements.
Rewards	match_reward - (movement_cost * movement_cost_factor) - total_over_allocated_penalty
Observation Space	For each grid zone, the observation includes the predicted need probability for each resource type and the current allocation of those resources.
Constraints	Limits on total resources (e.g., 50 police units) and maximum per-zone allocations (e.g., 8 units per type).
Learning Objective	Maximize long-term zone coverage while minimizing unnecessary movement, idling, and response delays.
Type of Resources	EMS (ambulance), Police, Fire Truck, Tow Truck
Type of Accidents	Property Damage, Injury (Minor, Severe, Fatal), Skyway (Highways)

Environment Setup



Buffalo City Accident Heatmap

The visualization highlights accident distribution across Buffalo, aggregated into a grid where each cell represents a 500m × 500m area in the real world. Darker regions indicate higher frequencies of incidents, effectively identifying accident hotspots.

Proximal Policy Optimization (PPO)

This algorithm improves an agent's behavior while keeping updates stable. It uses a clipped objective function to prevent the policy from changing too much in one step, which makes training more reliable.

Algorithm 1 PPO-Clip

Input: initial policy parameters θ_0, initial value function parameters ϕ_0
for $k = 0, 1, 2, \ldots$ **do**
  Collect set of trajectories $\mathcal{D}_k = \{\tau_i\}$ by running policy $\pi_k = \pi(\theta_k)$ in the environment.
  Compute rewards-to-go $\hat{R}_t$.
  Compute advantage estimates, $\hat{A}_t$ (using any method of advantage estimation) based on the current value function V_{ϕ_k}.
  Update the policy by maximizing the PPO-Clip objective:

$$\theta_{k+1} = \arg\max_{\theta} \frac{1}{|\mathcal{D}_k|T} \sum_{\tau \in \mathcal{D}_k} \sum_{t=0}^{T} \min\left(\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_k}(a_t|s_t)} A^{\pi_{\theta_k}}(s_t, a_t),\ \ g(\epsilon, A^{\pi_{\theta_k}}(s_t, a_t)) \right),$$

  typically via stochastic gradient ascent with Adam.
  Fit value function by regression on mean-squared error:

$$\phi_{k+1} = \arg\min_{\phi} \frac{1}{|\mathcal{D}_k|T} \sum_{\tau \in \mathcal{D}_k} \sum_{t=0}^{T} \left(V_\phi(s_t) - \hat{R}_t \right)^2,$$

  typically via some gradient descent algorithm.
end for

PPO Algorithm

PPO Implementation Details

PPO Implementation	
Action Space	• Policy network outputs parameters of a Gaussian distribution • Actions are sampled during training to enable exploration.
Advantage Calculation	• Generalized Advantage Estimation (GAE, λ = 0.95) used to compute advantages
Policy Optimization	• PPO clipped surrogate objective (clip_range = 0.2) used to restrict policy updates • Promotes stability during training.
Value Function	• Critic trained using Mean Squared Error (MSE) loss • Weighted by a value loss coefficient (vf_coef = 0.5).
Exploration Strategy	• Entropy bonus (ent_coef = 0.01) added to the loss to promote exploration • Prevents early policy convergence.
Training Setup	• 4 parallel environments (SubprocVecEnv) for efficient data collection (n_steps = 2048) • 10 update epochs per phase using Adam optimizer • Gradient clipping (max_grad_norm = 0.5) & GPU acceleration.

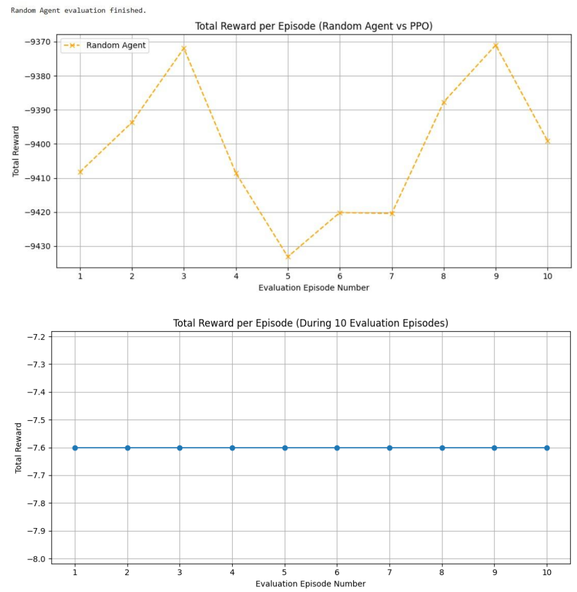
Architecture: An Actor-Critic model using Multi-Layer Perceptron (MLP) networks for both the policy (actor) and value function (critic). These networks processed the state vector comprising predicted resource needs and current allocations

Random Agent

The random agent allocates and deallocates resources arbitrarily across all grid blocks. It operates in the same environment with the same reward structure.

Results & Conclusion

This project demonstrated how reinforcement learning can optimize emergency resource allocation in a city like Buffalo. Using accident probability and traffic flow data, the PPO-trained agent consistently achieved a reward of -7.6 across 10 evaluation episodes. In contrast, the random agent's rewards ranged between -9430 and -9370, highlighting the PPO agent's ability to make informed and efficient allocation decisions in high-risk zones.



References:

https://spinningup.openai.com/en/latest/algorithms/ppo.html

https://data.buffalony.gov/Public-Safety/Received-Traffic-Incident-Calls/6at3-hpb5/about_data

University at Buffalo The State University of New York

Department of Computer Science & Engineering
School of Engineering & Applied Sciences
fagunnir@buffalo.edu | meghnatr@buffalo.edu | sahilshi@buffalo.edu