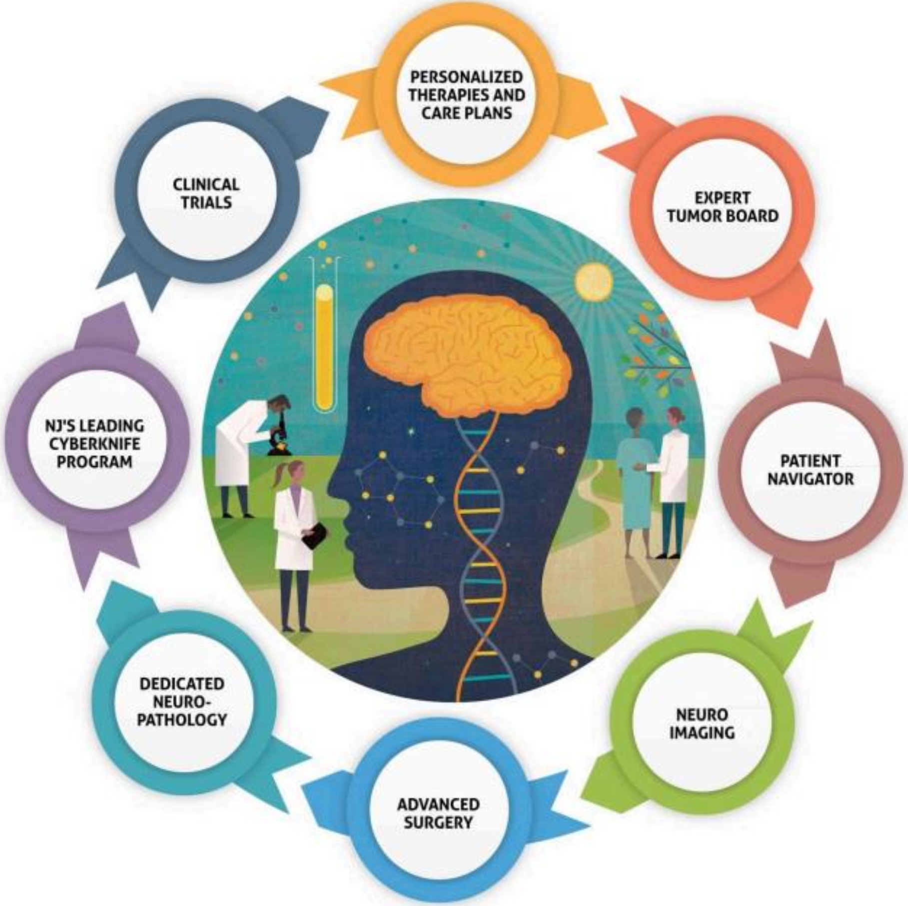
Brain Tumor MRI Classification and Localization

University of Information Technology (UIT) – VNU HCMC
AI – CS106
Quan Hoang Ngoc - 22521178
Cam-Giang Tran-Thi - 22520361
Professor: PhD. Hoang Luong Ngoc

6/2024

Table of Contents

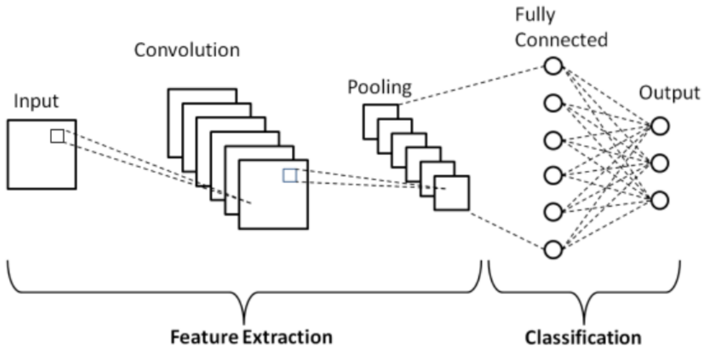
Table of Contents



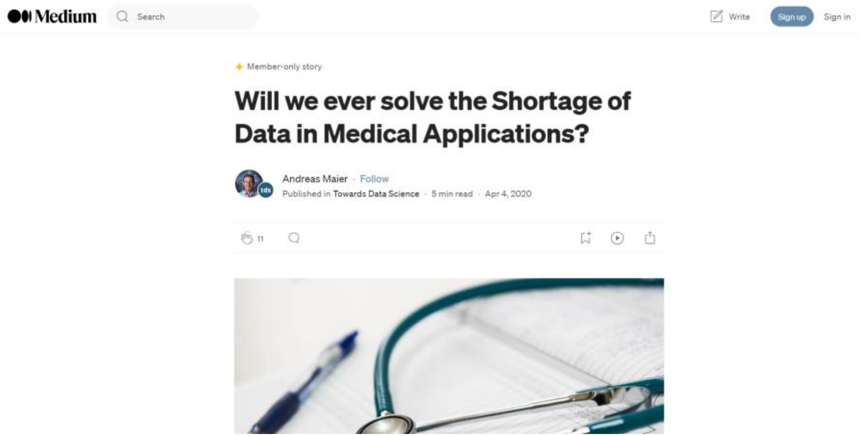
Introduction

Why it matters?

Brain tumor classification and localization: is a crucial task in medical imaging for accurate diagnosis and treatment planning.



Schematic diagram of a basic convolutional neural network (CNN) architecture



CNNs model are very robust to address with **large amounts of training data**.

But, more and more **scarcity of medical data**. How to deal these challenges?

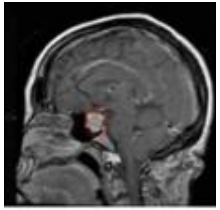
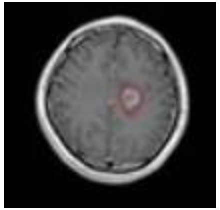
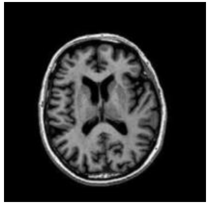
Problem

Input:	**Output**:
• Inference: a brain MRI image, which be a 2D slice. • Training: a **shortage** dataset consists of labeled images with the corresponding annotation for each image.	• The output of the model is a **predicted label** (yes or no tumor). • Besides, we also provide the predicted **location of the tumor center**, as an evidence for the model's decision.

Constraint: this is a **few short learning** problem.

Problem

Inference illustration

	Yes (63, 50)
	Yes (50, 65)
	No

Contribution

The aim of research project

The aim of this research project is to develop methods can accurately classify and provide evidence of the center of the tumor that dealing good with scarcity of data.

Besides, we also perform a comprehensive comparison of various strategies and evaluate their effectiveness on shortage data benchmarks. Then, we try to explain these results.

Finally, we find out some novel insights and directions in Reinforcement Learning.

Methodology

History of project

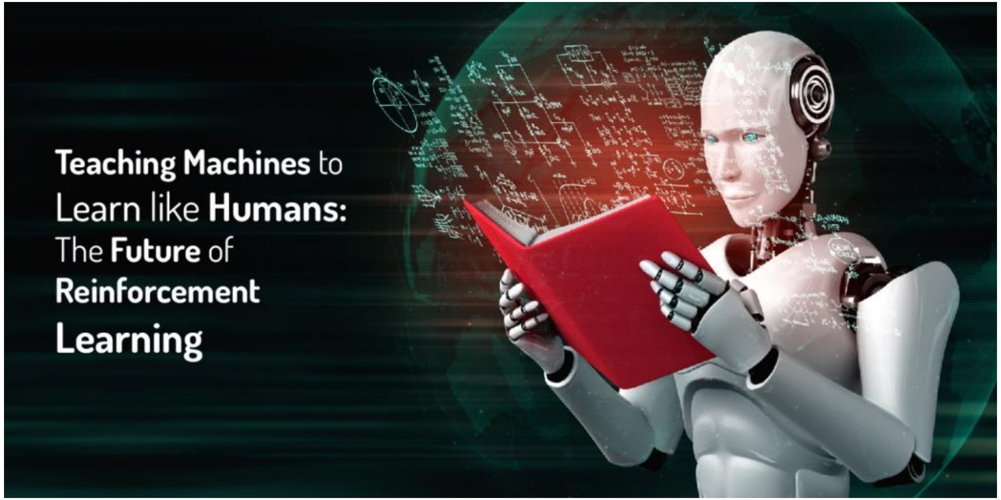
In all this project,

- We have come up with and applied a variety of strategies and techniques, including Machine Learning, Computer Vision, Reinforcement Learning, and combine them.
- However, within the limitation of time, we can only focus on breakthrough solutions and present the highlights of the project.

The breakthrough

Reinforcement Learning

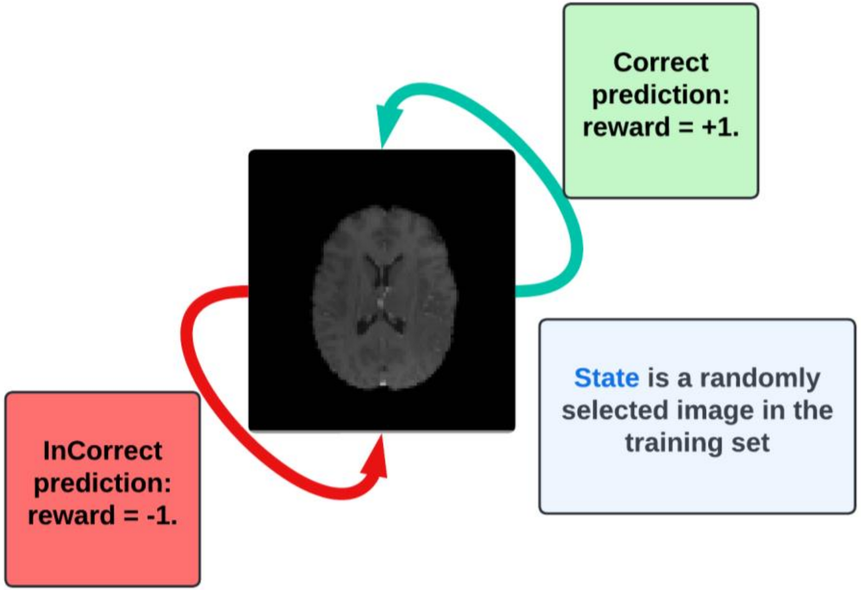
Hypothesis: It is our conviction that Reinforcement Learning (RL) will exhibit similarities to **human** learning by leveraging a minimal amount of information to generalize across differences. Through a process of insight search, exploration and exploitation of the training dataset, RL can efficiently navigate limited data spaces.



Classification

How to define a MDP

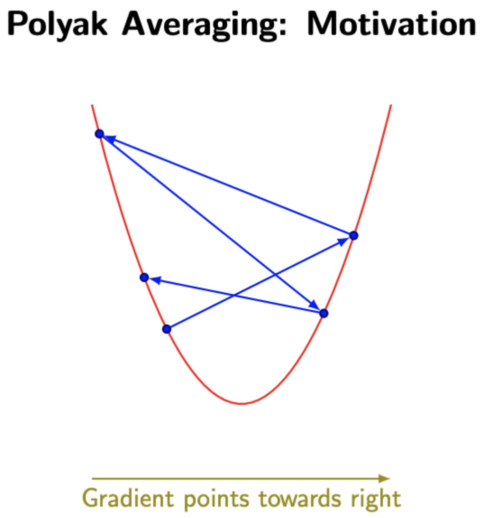
Action: An agent's action is a **label prediction** for the image which is a **state** of the environment, where 0 corresponds to no, and 1 corresponds to yes.



Classification

The algorithm: Momentum DQN

Polyak Averaging (Exponential Moving Average): is a technique used to optimize parameters in certain mathematical algorithms. The idea is to take the average of recent parameter values and set the final parameter to that average. The purpose is to help algorithms converge to a better final solution.



Classification

The algorithm: off-policy greedy strategy

$$a_t = \begin{cases} max_{a\epsilon A}\{Q_t(a)\} & with\ probability\ \epsilon \\ random\ action\ in\ A & with\ probability\ 1 - \epsilon \end{cases}$$

$$\varepsilon = 1.0 \rightarrow 0.01, \qquad \varepsilon\_decay = 0.995$$

Classification

The algorithm: Q computation

Q definition

$$Q^{\pi}(s,a) = \mathrm{E}_{\pi}\{R_t | s_t = s, a_t = a\}$$

$$= \left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s, a_t = a\right\}$$

Target computation:

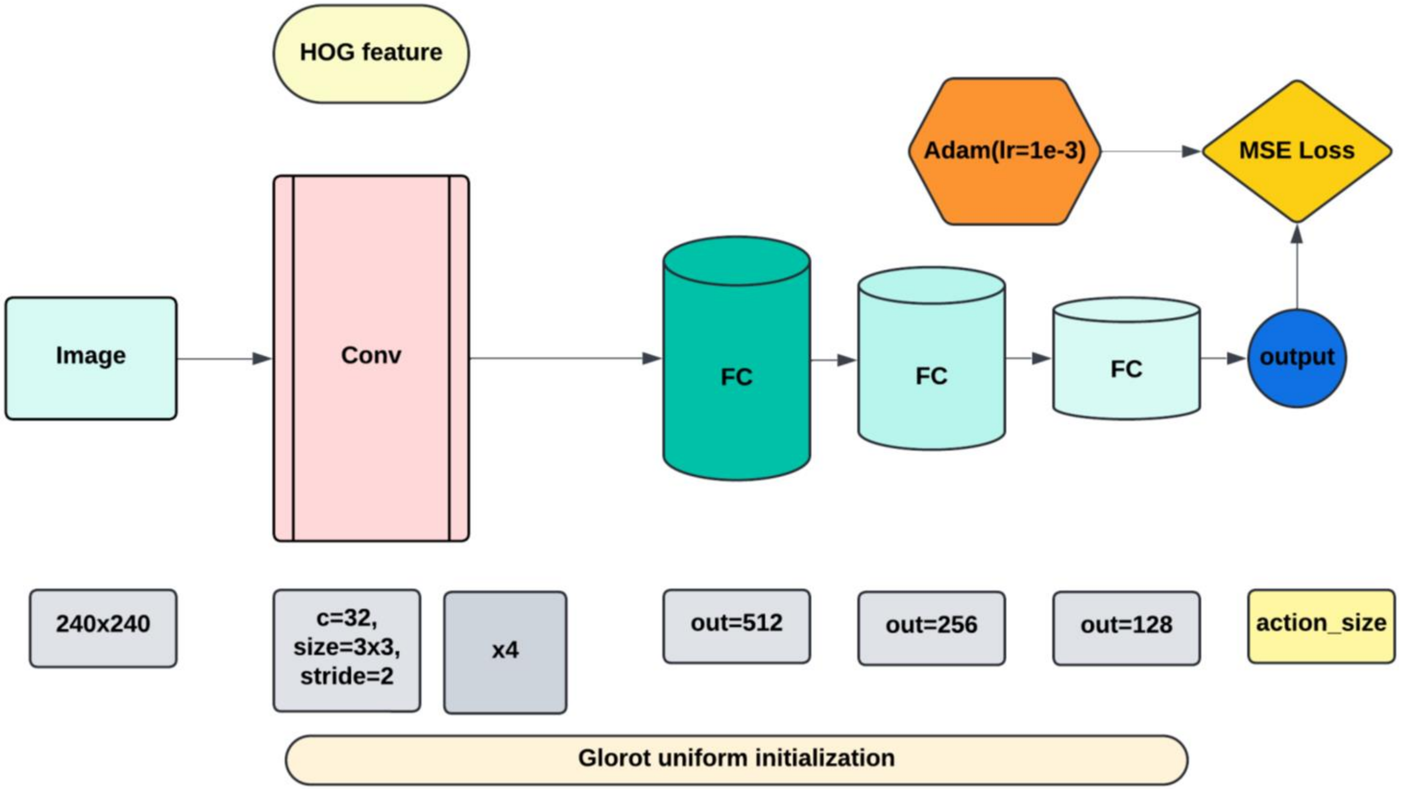
$$Q_{target}^{(t)} = r_t + \gamma max_a Q(s_{t+1}, a),$$

Prediction computation:

$$Q_{DQN}^{(t)} = F_{DQN}(s_t), \qquad \lim_{t \to \infty}\left(Q_{DQN}^{(t)}\right) = \lim_{t \to \infty}\left(Q_{target}^{(t)}\right) = Q^*$$

Classification

Deep Q Network Architecture



Classification

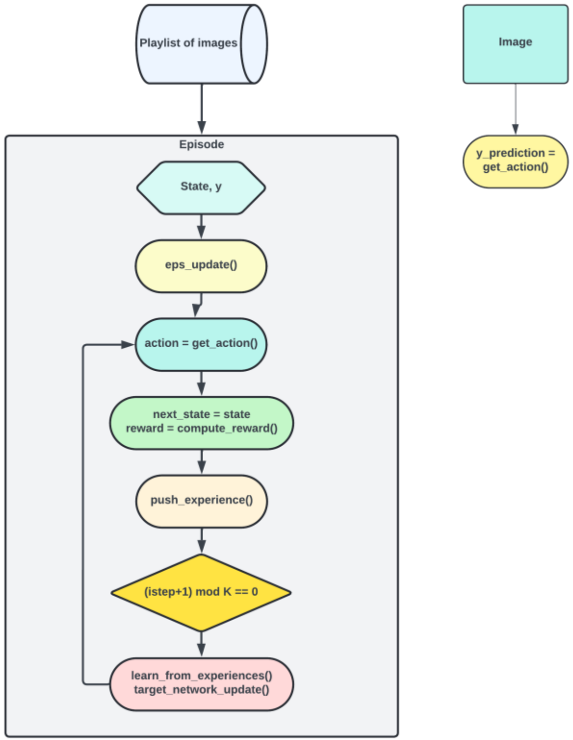
Network Update

After each ***K*** steps, both **local and target network** will be updated. The local network will be learned from experiences and the target network parameters will be updated with **Polyak averaging**.

$$p_{target} = \tau \times p_{local} + (1 - \tau) \times p_{target}$$

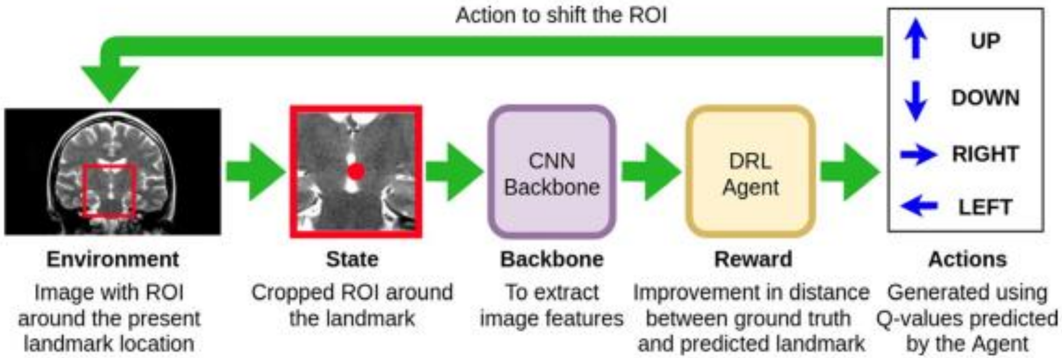
Classification

Flowchart



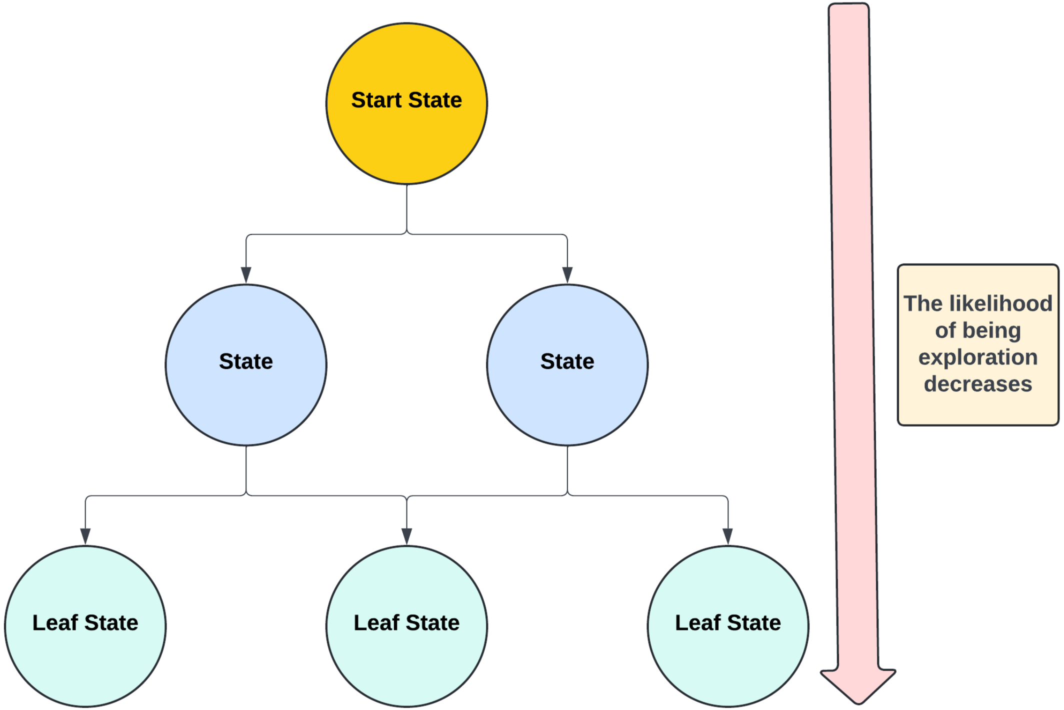
Localization

Traditional Approach



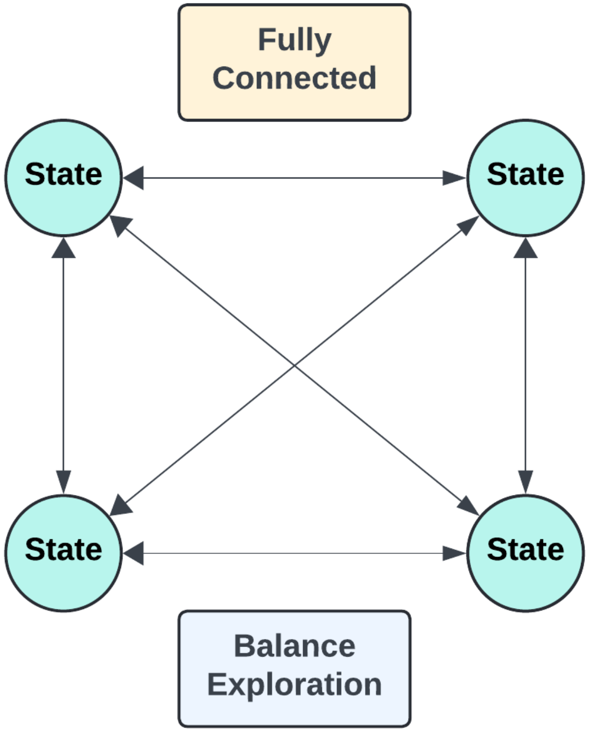
Localization

Traditional Limitation: DAG Environment



Localization

Finding: Complete Graph

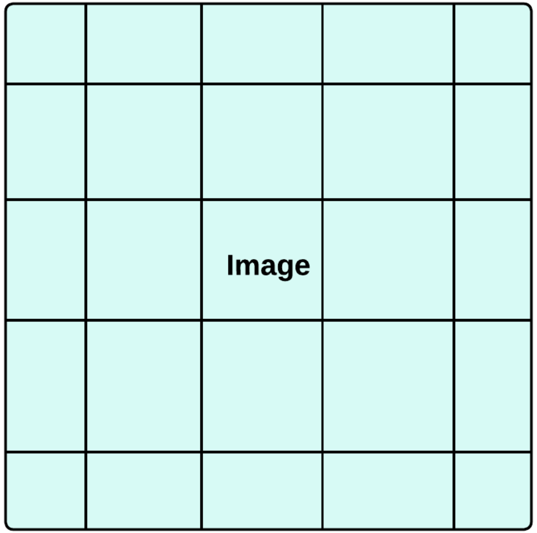


Localization

Finding: The breakthrough

Idea: Predicting the location of the tumor center instead of moving the bounding box to the tumor

Infinite action space: each action = a pixel of image (state)



State

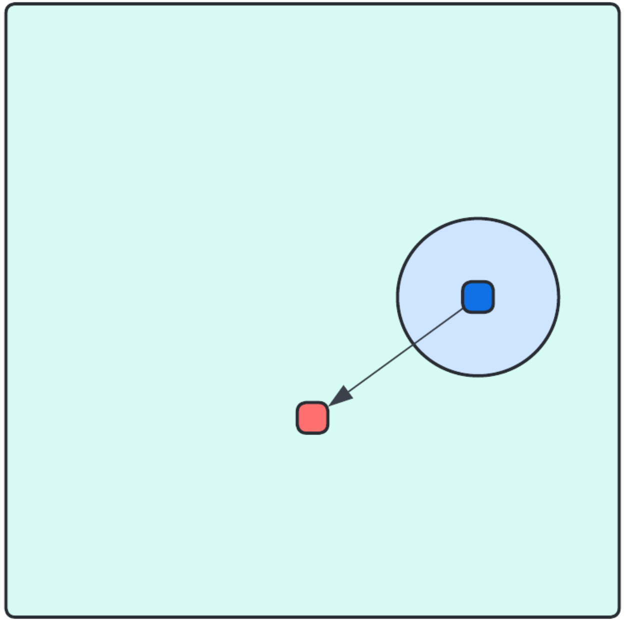
From **Localization** to **Regression**

YOLO

You Only Look Once : Unified , Real-Time Object Detection

Localization

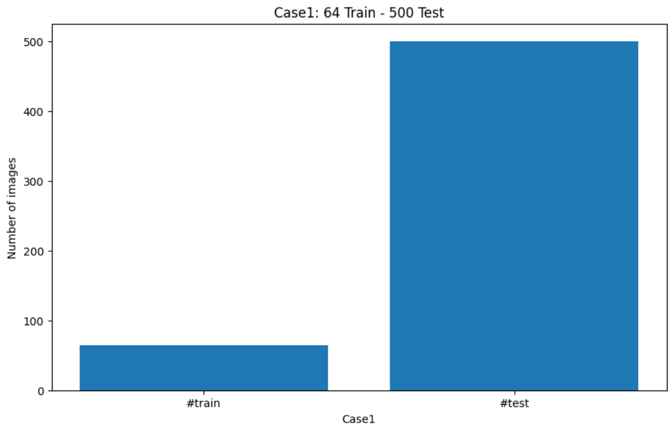
Finding: Reward Computation

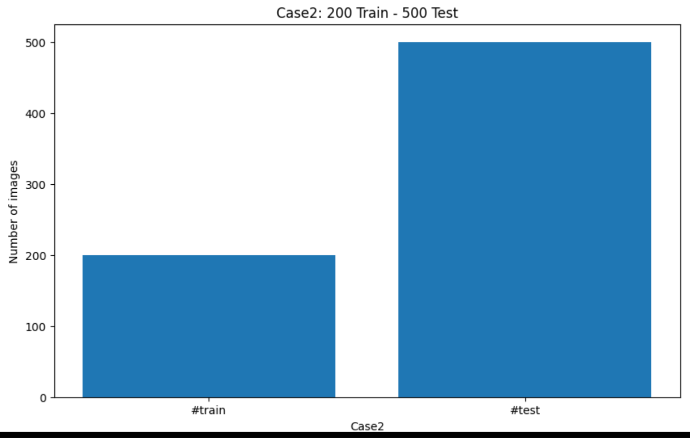


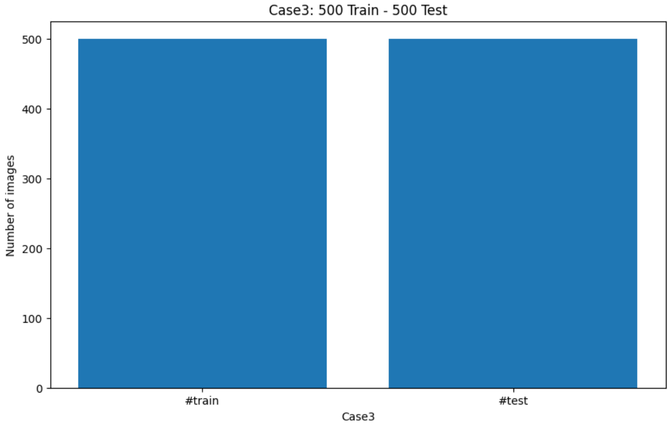
In correct radius: +1.
Out radius: -0.5 * distance / radius

Comprehensive Experiment

Multi case: classification

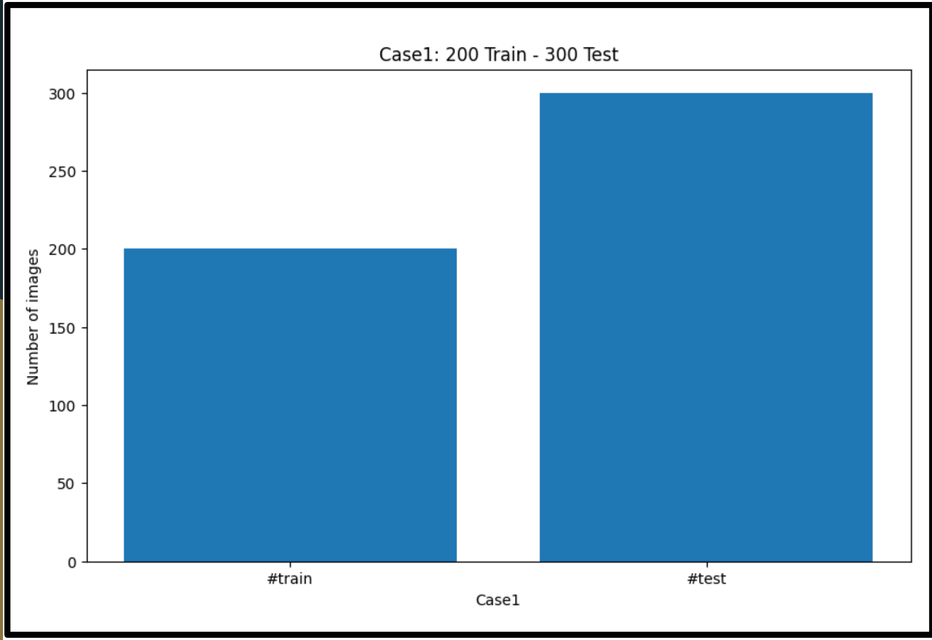


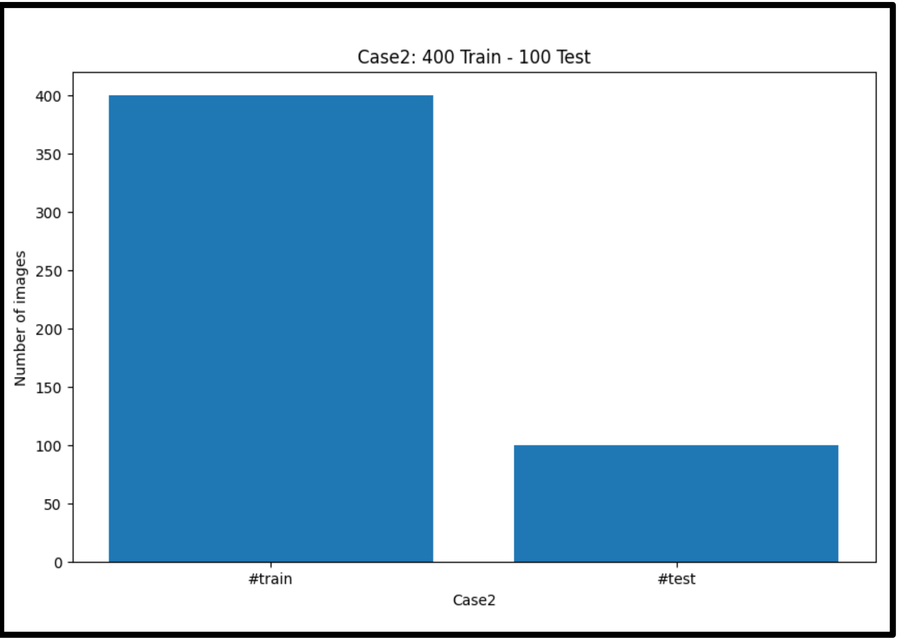




Comprehensive Experiment

Multi case: localization





Comprehensive Experiment

Multi baseline

Machine Learning: SVC, KNN
Deep Learning: VGG16
Reinforcement Learning: Deep Q Network and Momentum DQN

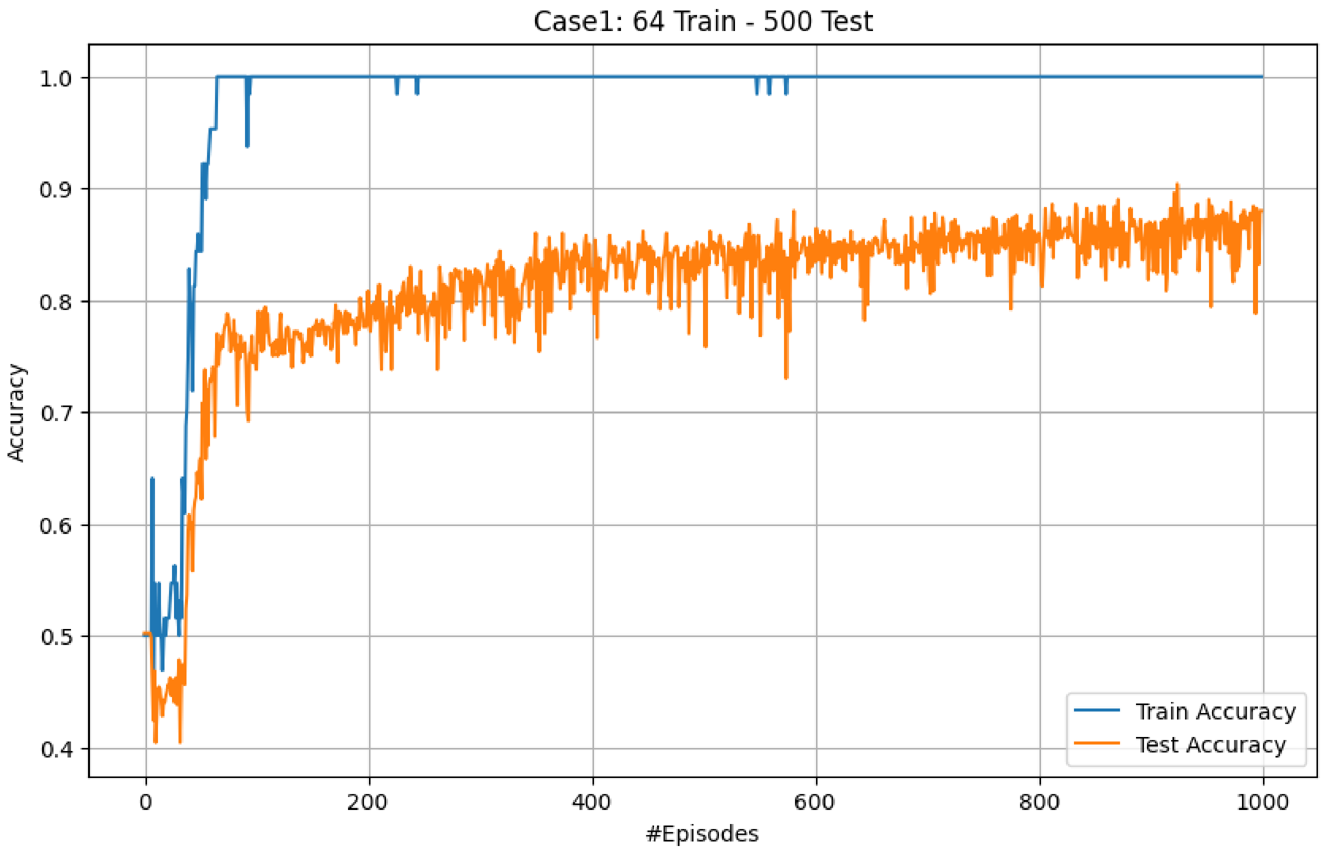
Experiment Result (1)

ML and DL

	Metric	KNN	SVC	VGG16
64-500	Accuracy	0.8600	0.8800	0.6580
	F1-score	0.8600	0.8800	0.6557
	Recall	0.8600	0.8800	0.6580
200-500	Accuracy	0.8700	0.9000	0.9120
	F1-score	0.8700	0.9000	0.9119
	Recall	0.8700	0.9000	0.9120
500-500	Accuracy	0.9300	0.9300	0.9440
	F1-score	0.9300	0.9300	0.9440
	Recall	0.9300	0.9300	0.9440

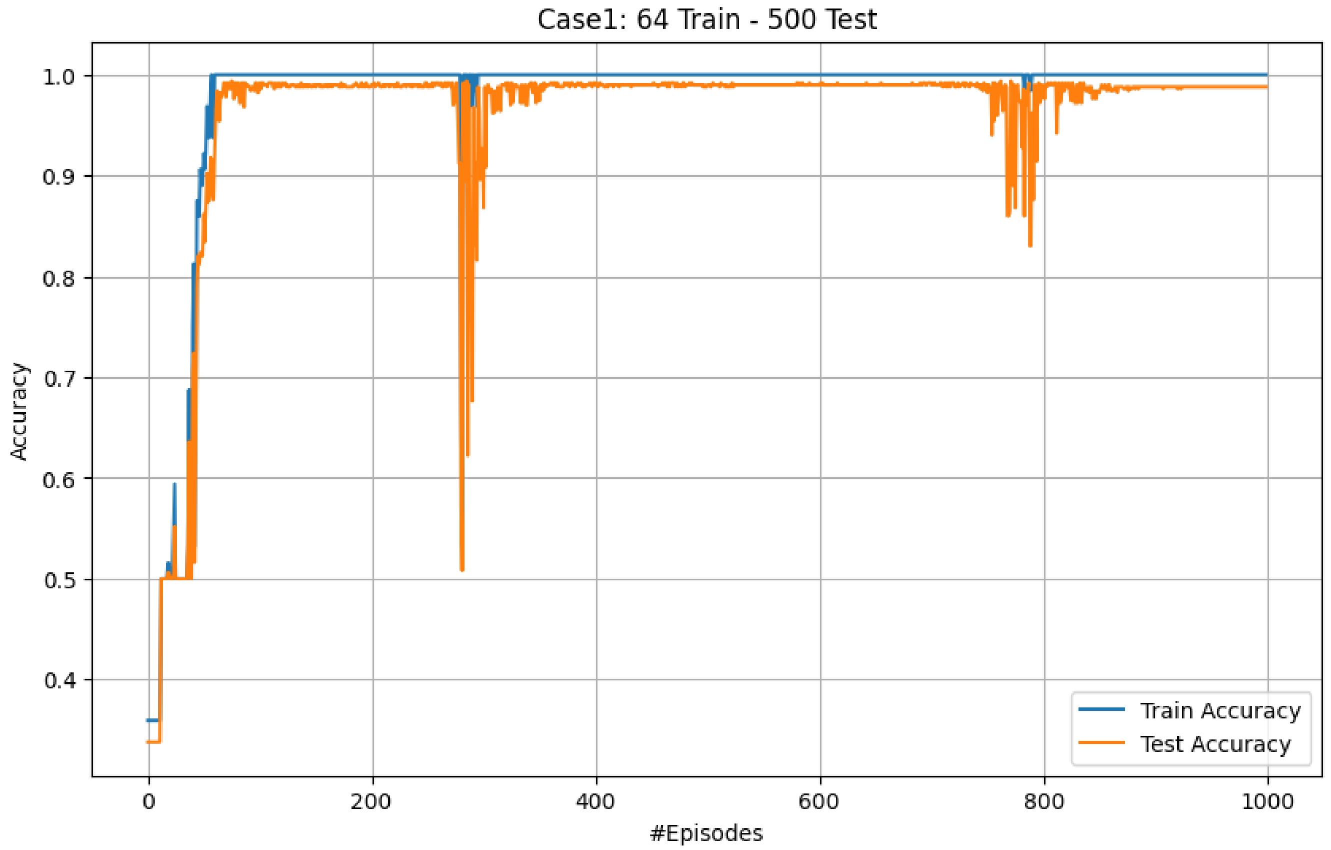
Experiment Result (2)

DQN



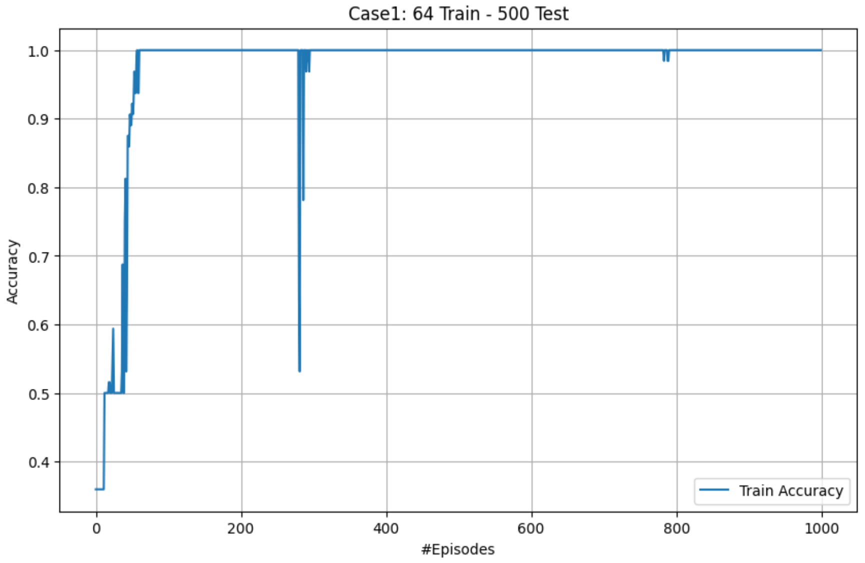
Experiment Result (2)

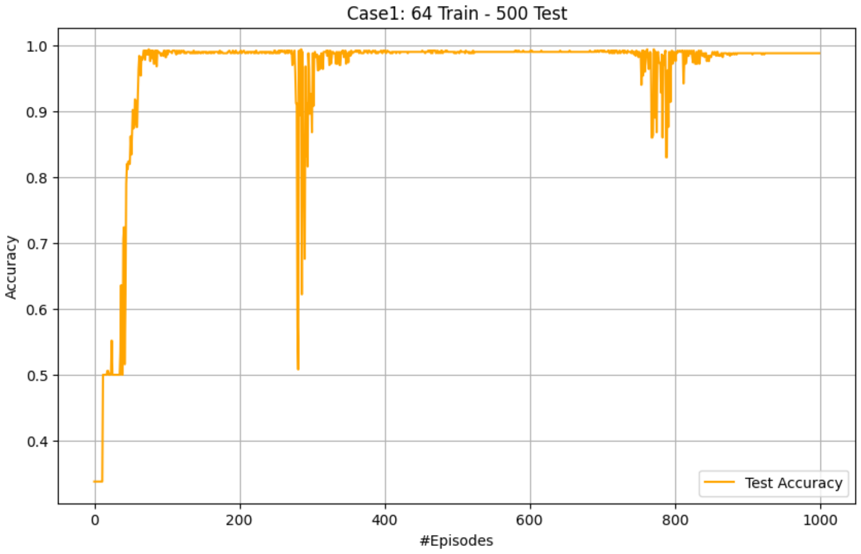
Momentum DQN



Experiment Result (2)

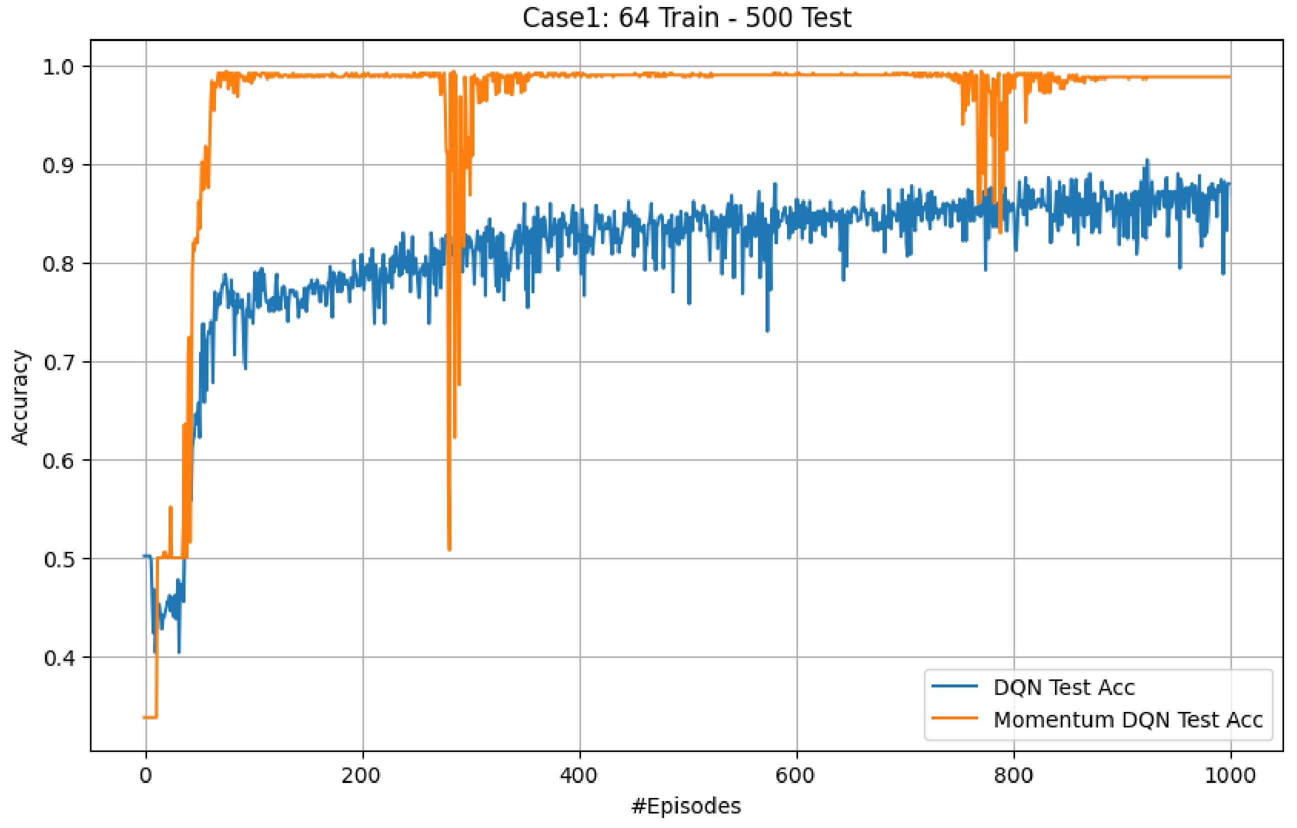
Momentum DQN





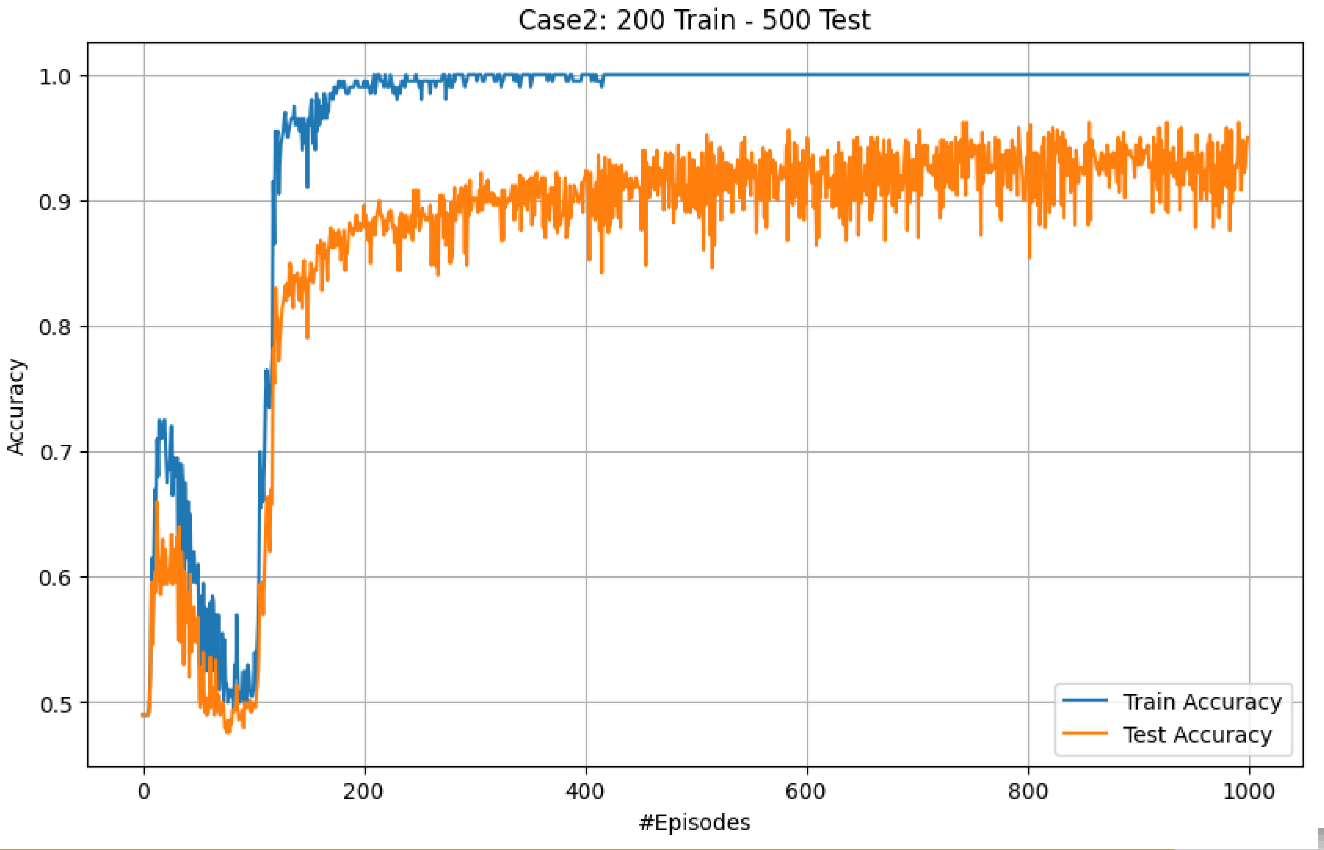
Experiment Result (2)

DQN vs. Momentum DQN



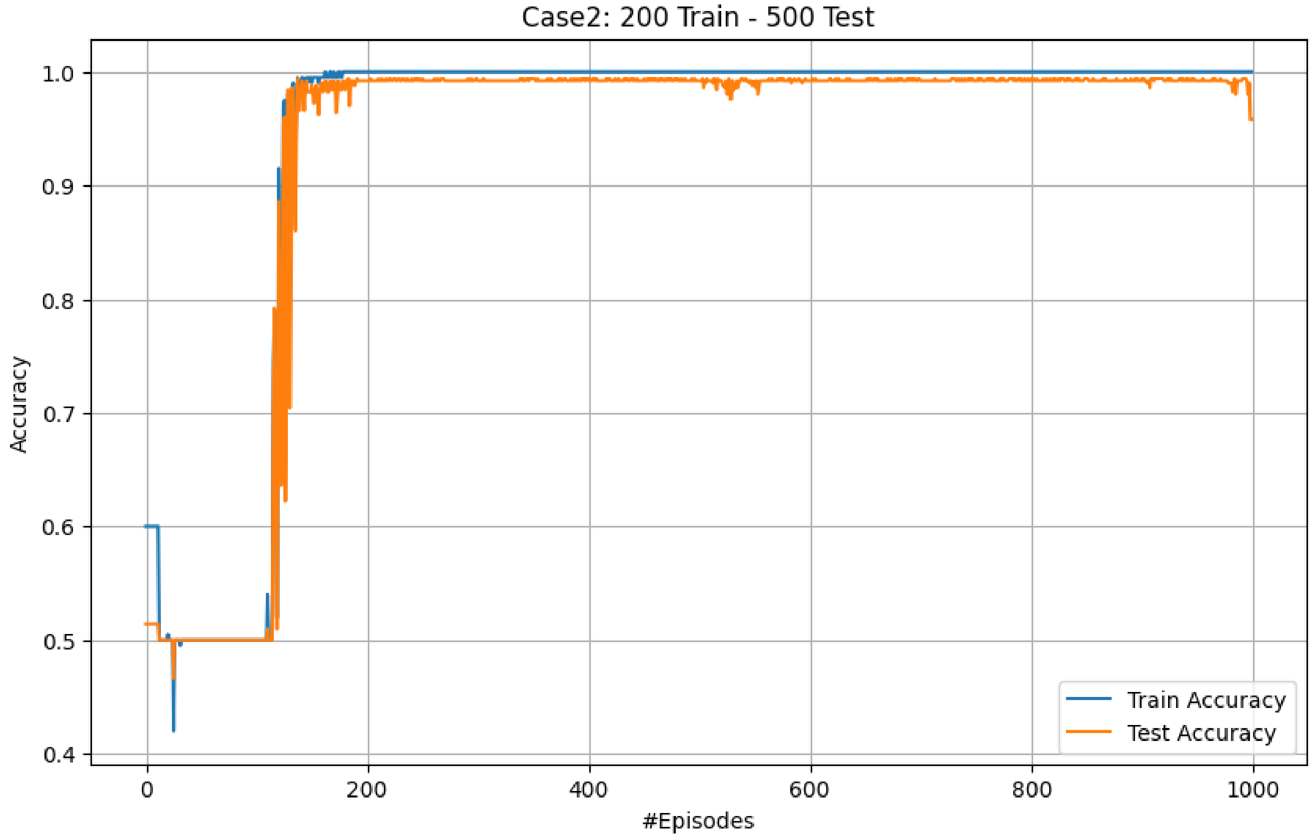
Experiment Result (3)

DQN



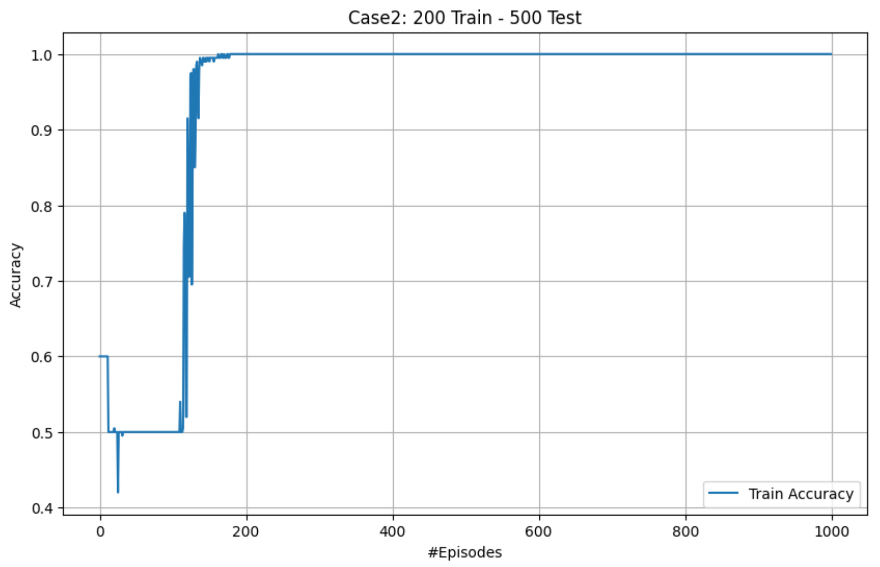
Experiment Result (3)

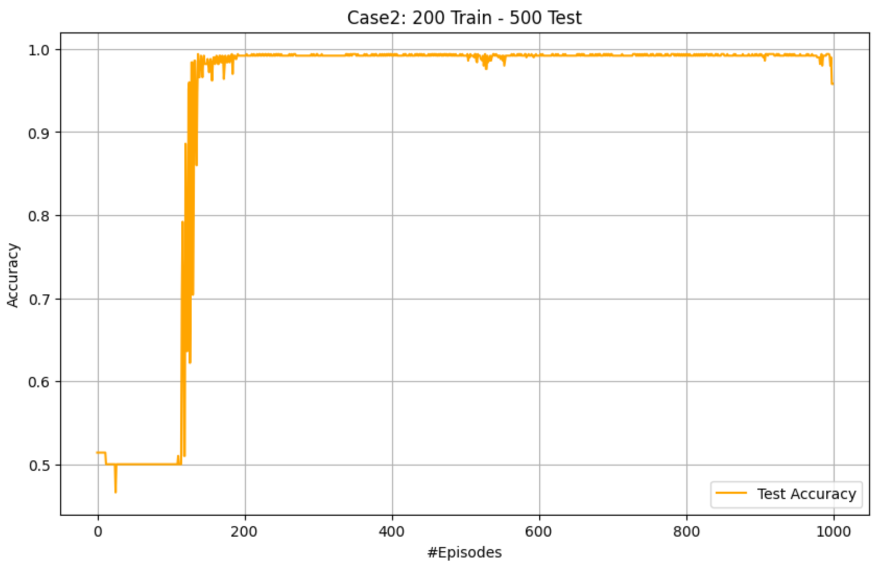
Momentum DQN



Experiment Result (3)

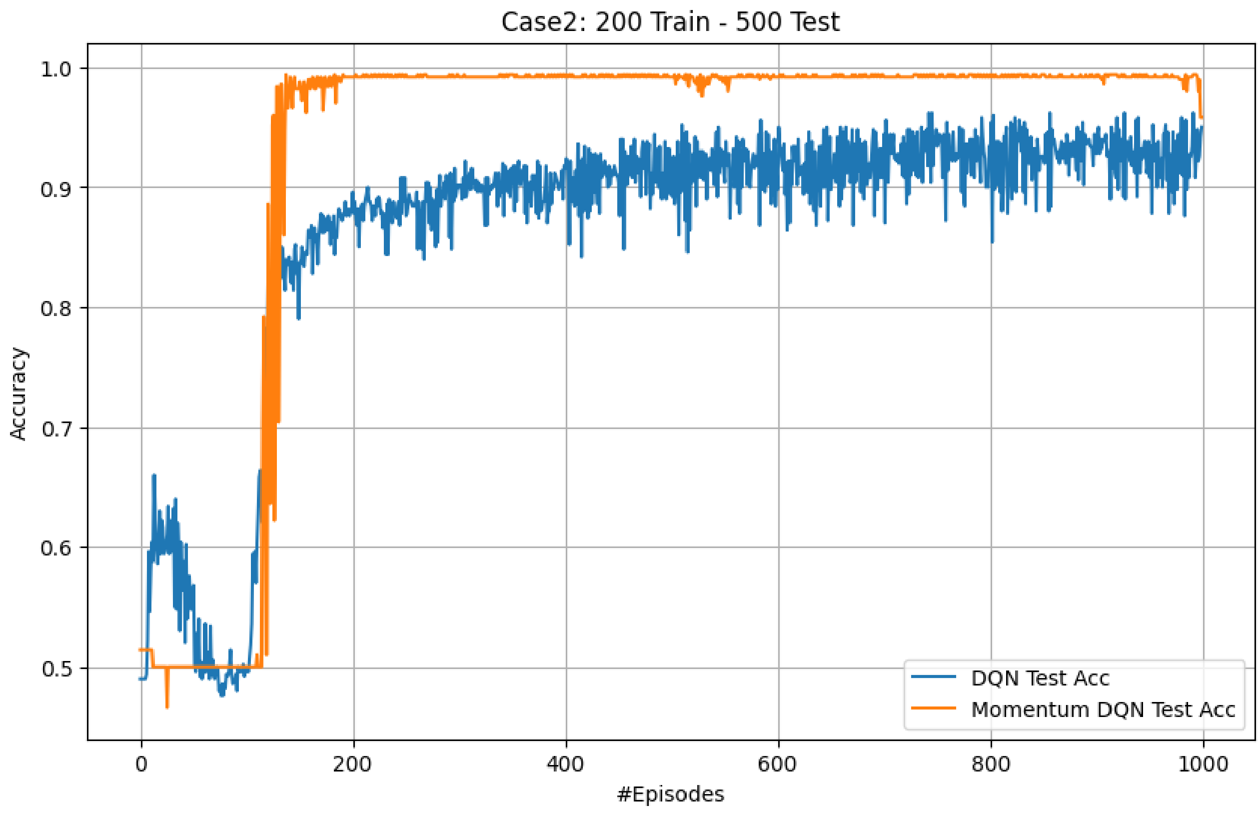
Momentum DQN



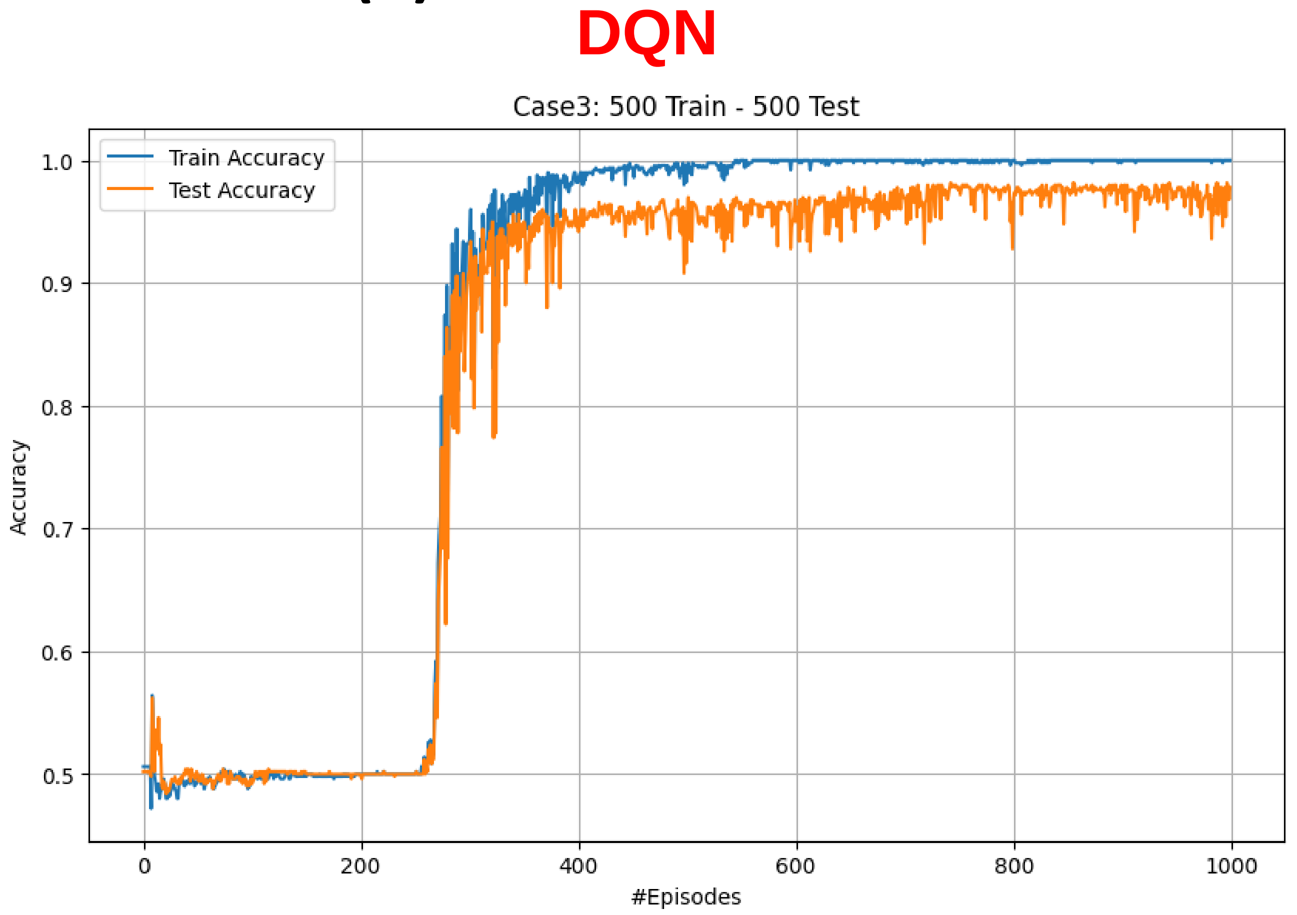


Experiment Result (3)

DQN vs. Momentum DQN

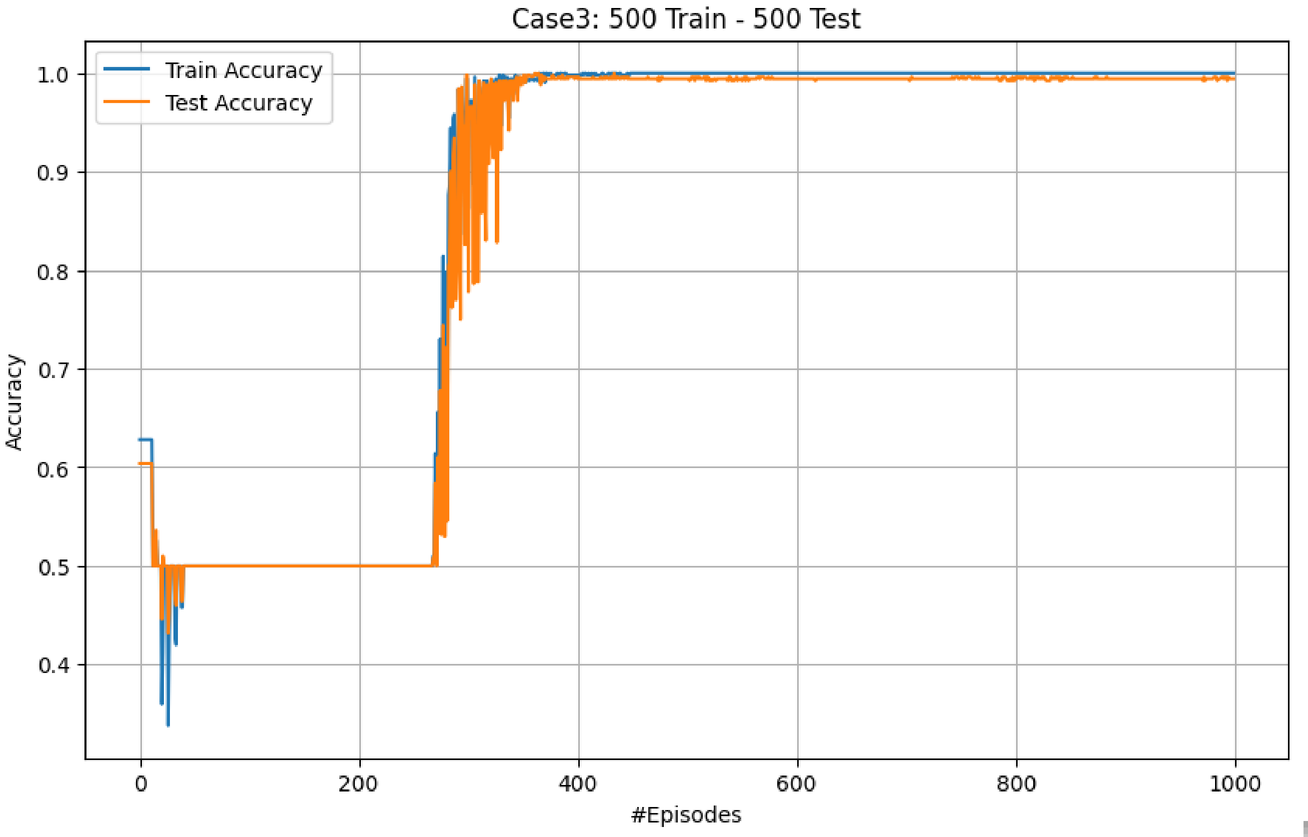


Experiment Result (4)



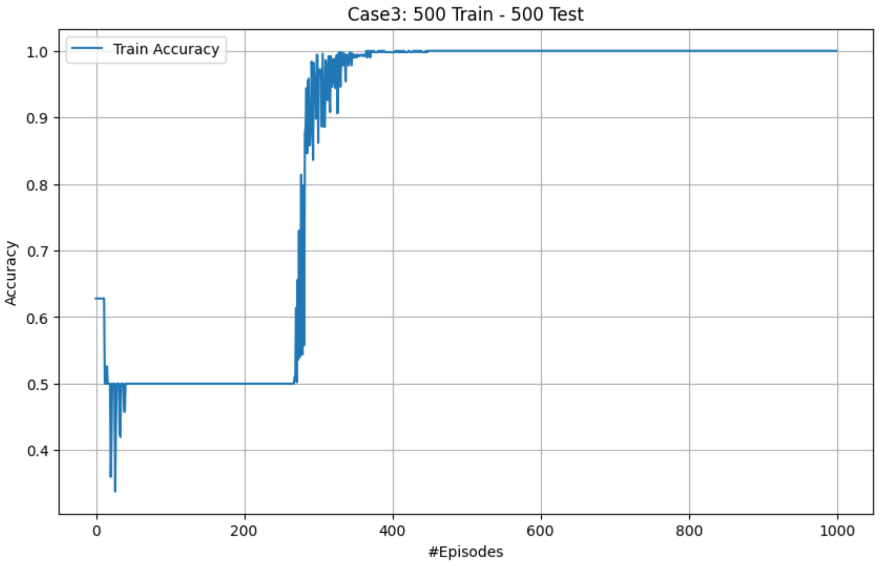
Experiment Result (4)

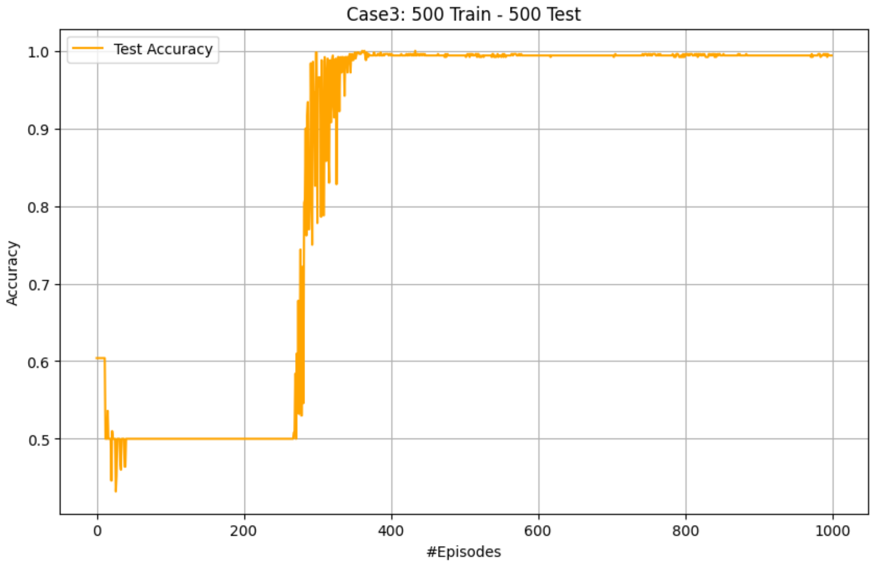
Momentum DQN



Experiment Result (4)

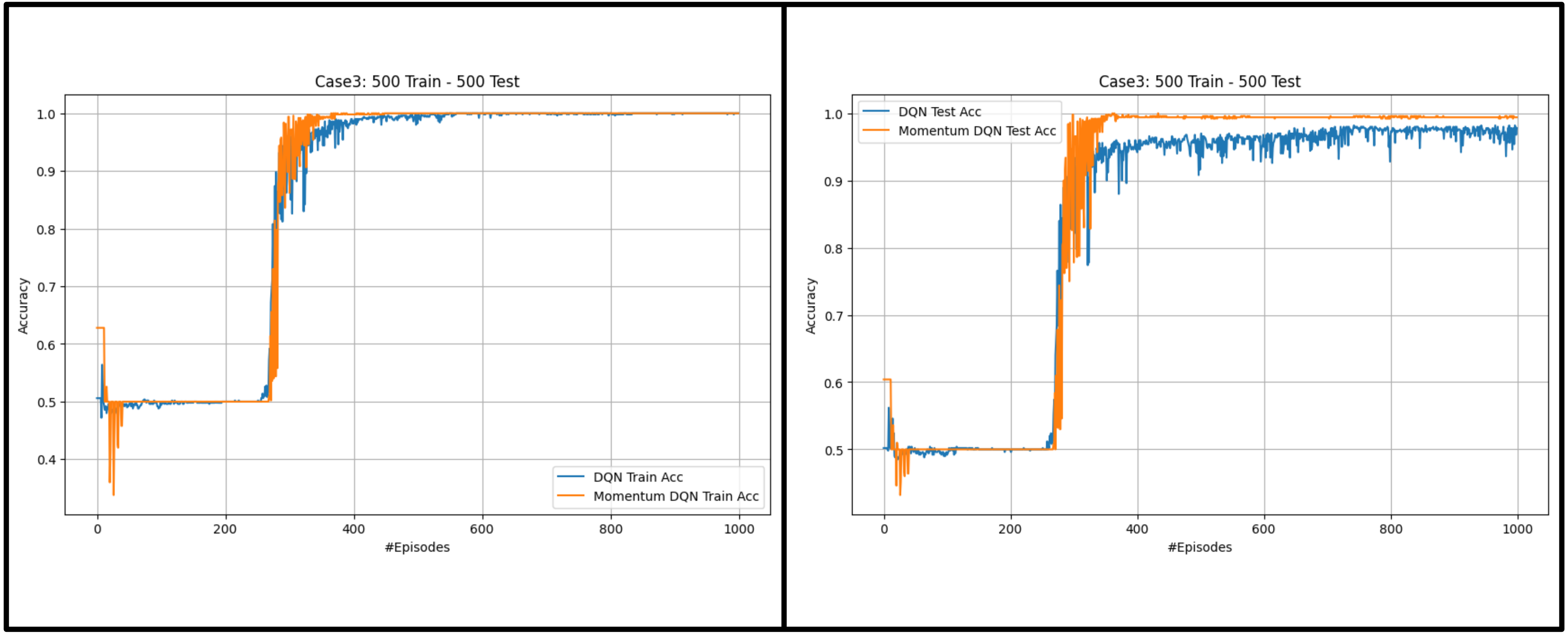
Momentum DQN



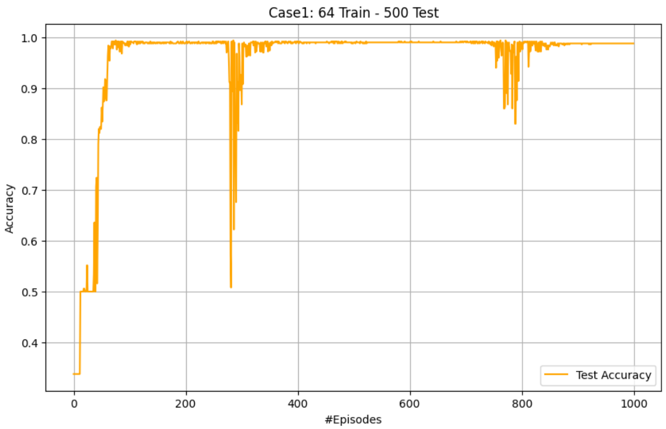


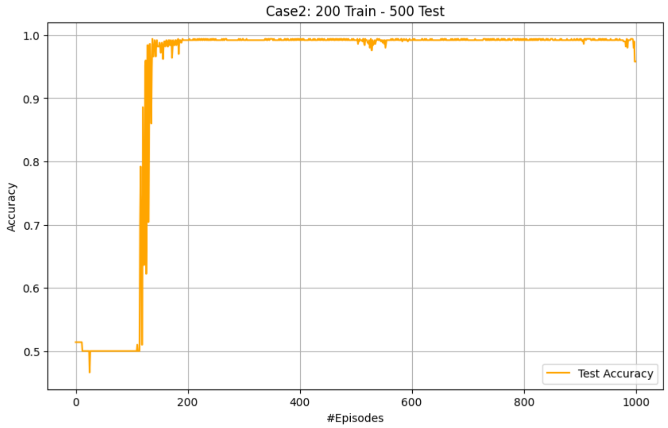
Experiment Result (4)

DQN vs. Momentum DQN

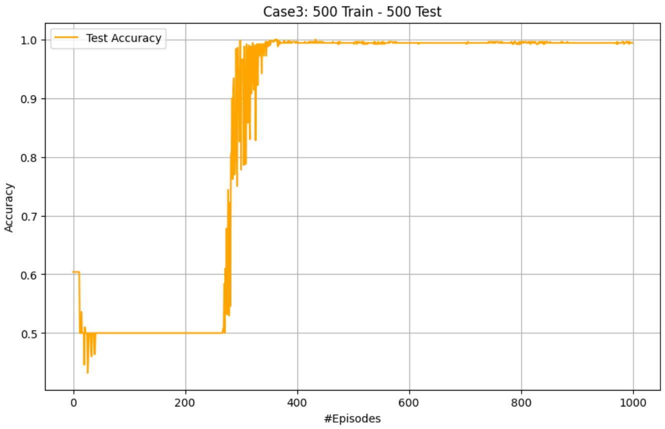


Result Experiment (5)



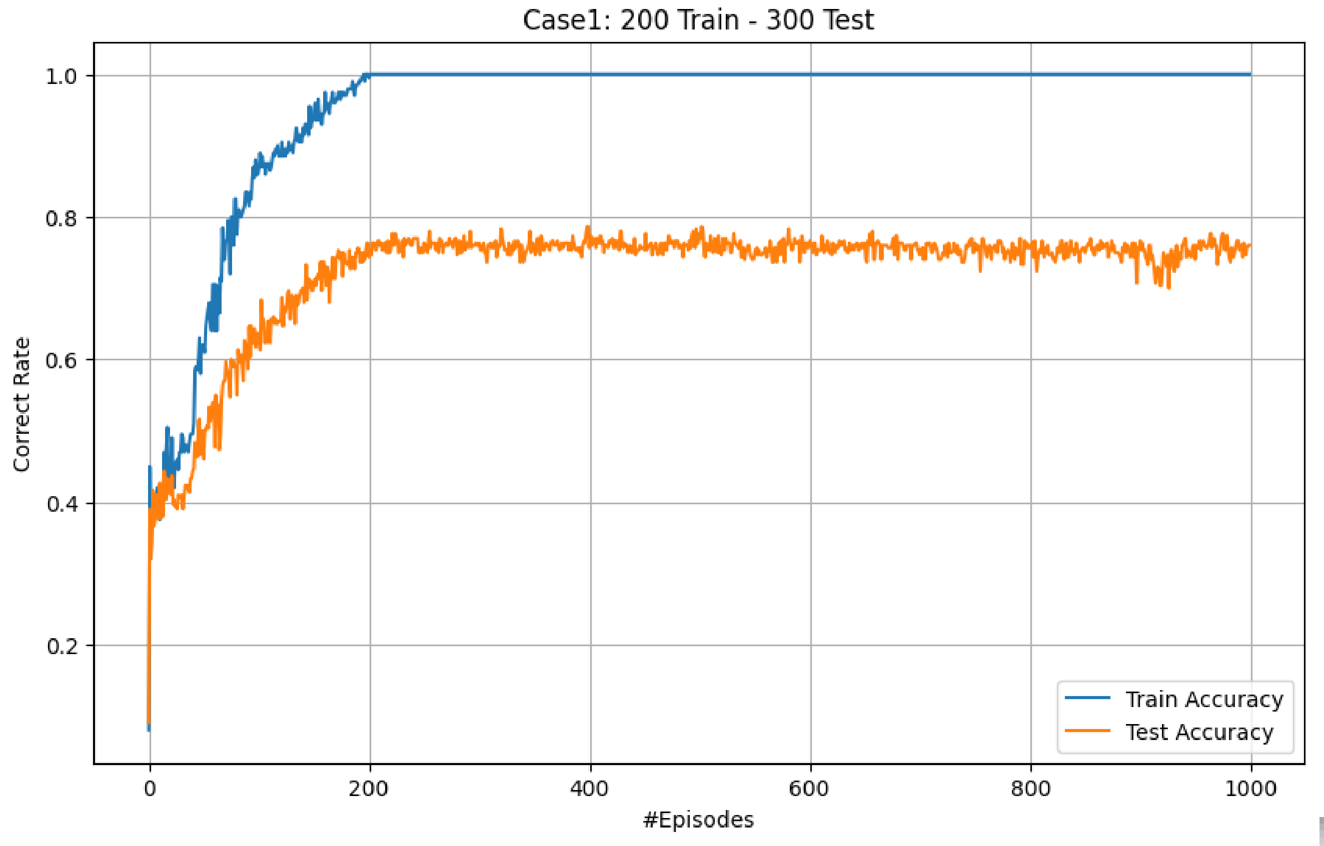


Momentum DQN Test Accuracy



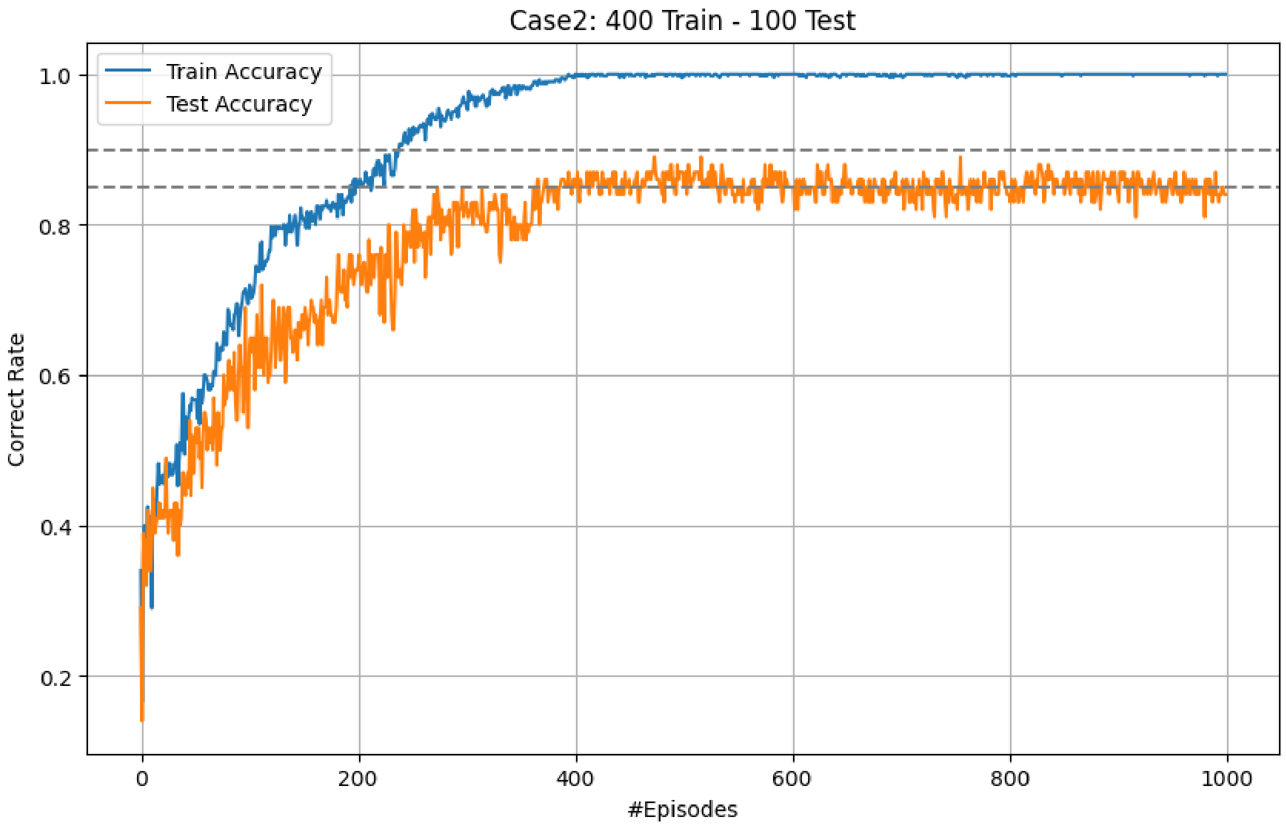
Experiment Result (6)

Localization: Momentum DQN



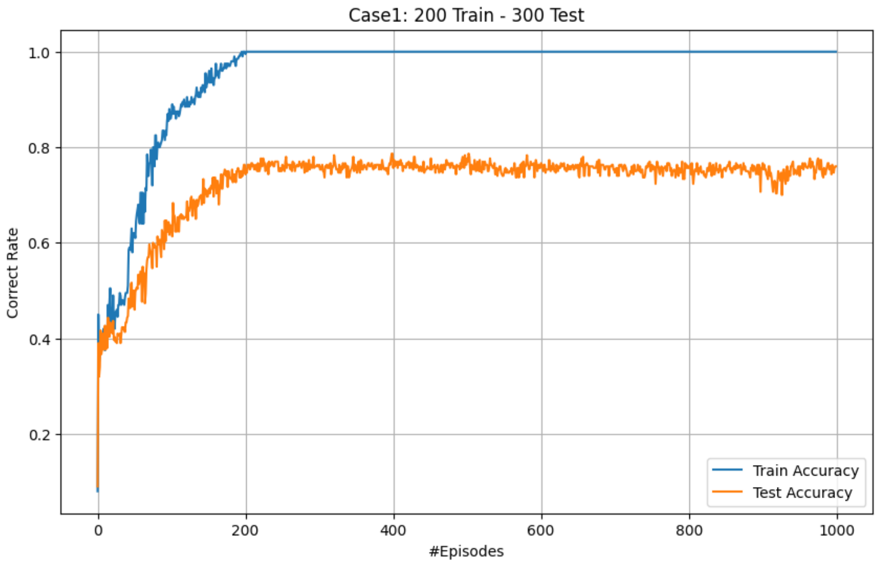
Experiment Result (6)

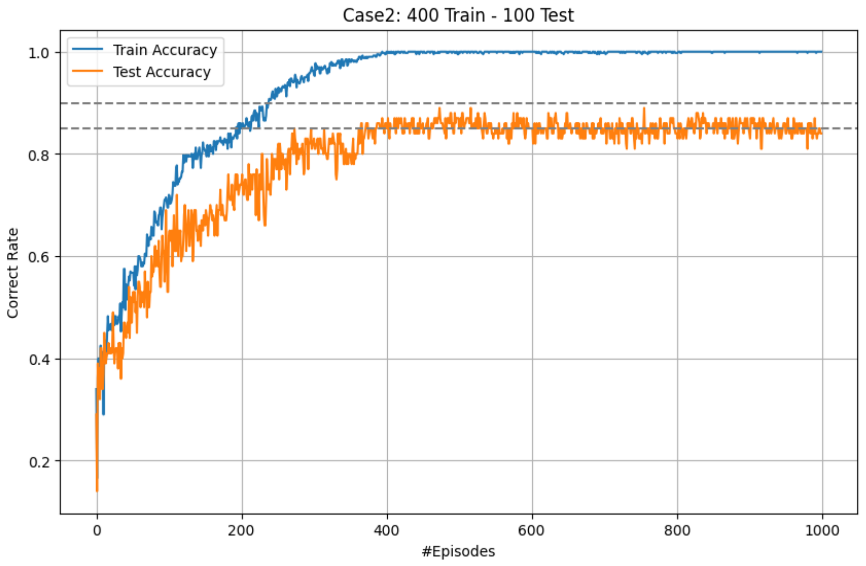
Localization: Momentum DQN



Experiment Result (6)

Localization: Momentum DQN





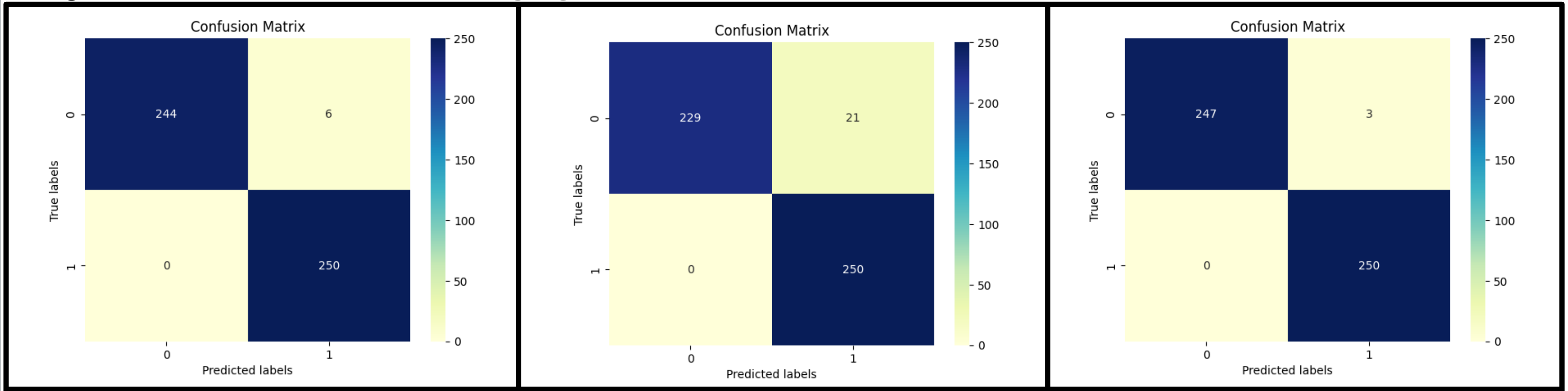
Experiment Result (7)

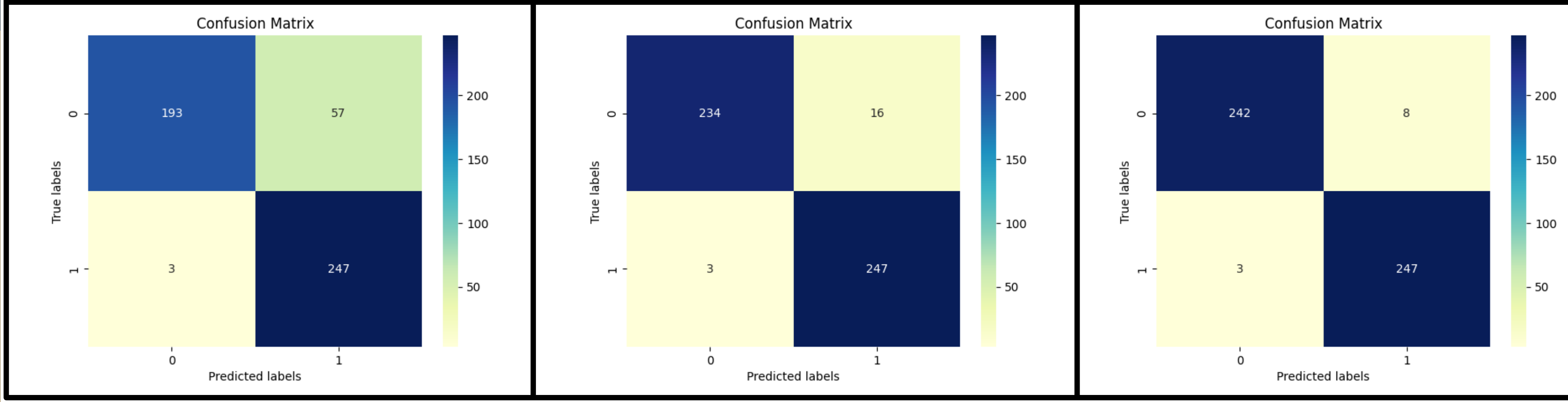
Comprehensive result

	Metric	KNN	SVC	VGG16	DQN	Momentum
64-500	Accuracy	0.8600	0.8800	0.6580	0.9040	0.9880
	F1-score	0.8600	0.8800	0.6557	0.9039	0.9880
	Recall	0.8600	0.8800	0.6580	0.9040	0.9880
200-500	Accuracy	0.8700	0.9000	0.9120	0.9620	0.9580
	F1-score	0.8700	0.9000	0.9119	0.9620	0.9579
	Recall	0.8700	0.9000	0.9120	0.9620	0.9580
500-500	Accuracy	0.9300	0.9300	0.9440	0.9820	0.9940
	F1-score	0.9300	0.9300	0.9440	0.9820	0.9940
	Recall	0.9300	0.9300	0.9440	0.9820	0.9940

Experiment Result (8)

Momentum DQN Confusion Matrix

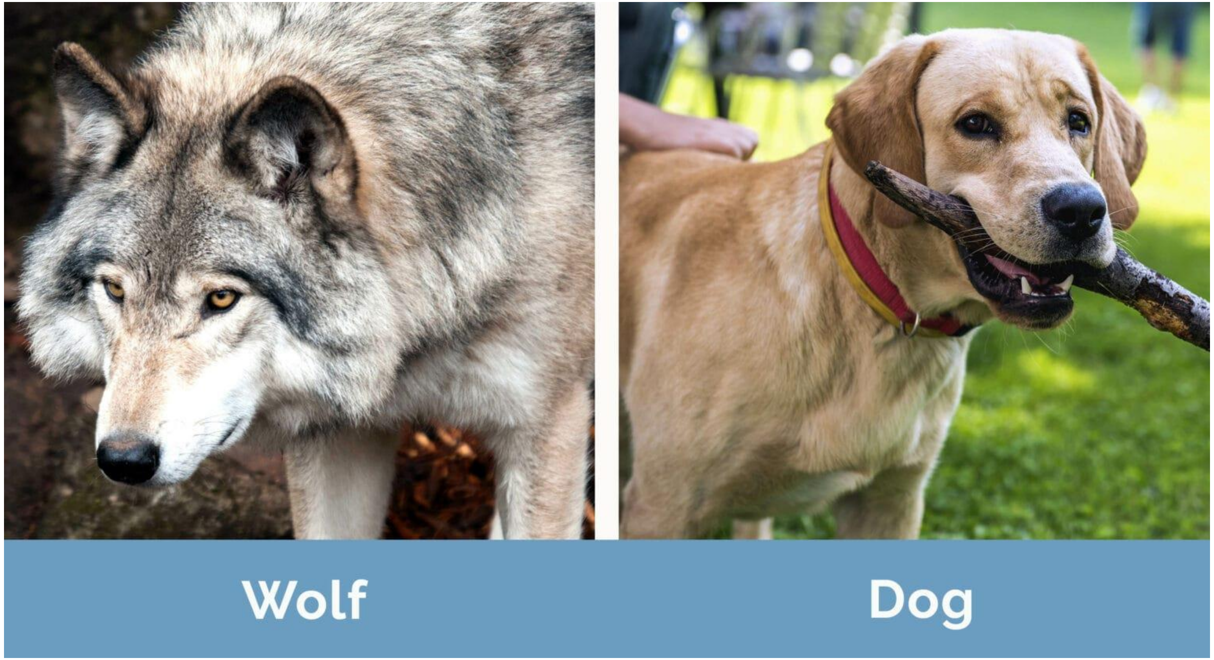




DQN Confusion Matrix

Explanation

How RL learn like human?
How CNNs learn?



Research Finding

Conclusion

- We found the effectiveness of Reinforcement Learning in solving data scarcity and give an explanation for that effectiveness.
- We also introduce the combination of DQN with **Polyak Averaging Update** which can improve model stability and learning (exploitation and exploration) performance.
- Localization can face many limitations with a grid environment, we also propose a novel approach and reveal some new directions for Reinforcement Learning in the future.
- In reality, having a variety of solutions to meet the diverse constraints of the problem is a crucial. Diverse solutions help us trade off, combine or improve solutions.
- Amid Reinforcement Learning is not a silver bullet and may face limitations when operating in isolation. Combining it with other simple techniques is a way to unleash its full potential.

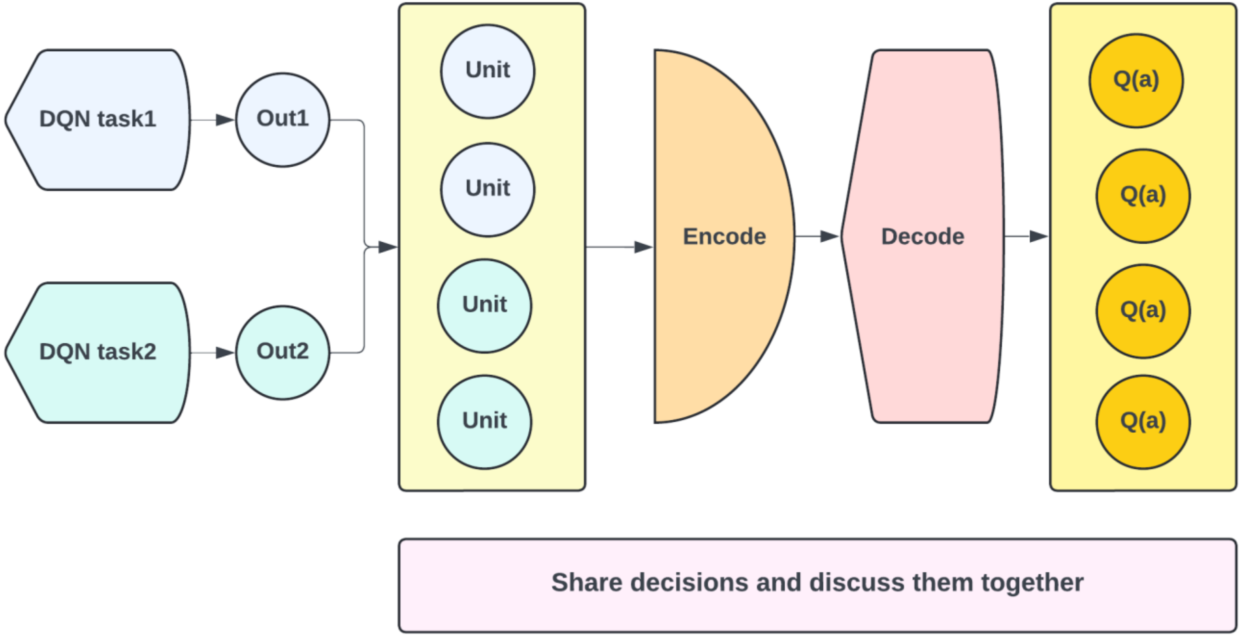
Recommendation

New Insight Finding

- Yolo version Reinforcement Learning?
- How to coordinate multi agents to perform multi tasks? – Auto Encoder – Fusion Model

Recommendation

Auto Encoder – Novel Fusion Model



Thank you

THANKS