# Federated Reinforcement Learning

**Ahmed Abuelyazeed** (*Media Engineering Technology*) and **Prof. Mohammed Salem** (*Digital Media Engineering*)

**The German university in Cairo**

Ahmed.abuelyazeed@student.guc.edu.eg

**Machine Learning(ML) applications have been a part of our everyday lives, from social media to self-driving cars and facial recognition. ML has significantly enhanced our quality of life in countless ways. Automation, smart decision-making, and improvement in healthcare are just a few of the promises of ML. Nevertheless, as with any new technology, the possibilities of ML are accompanied by a variety of challenges for society. ML models require access to a significant amount of data. This poses a serious privacy risk. A vast amount of data is often collected and processed by organizations. This data may be sensitive and is vulnerable to being exposed through data breaches. Federated Learning(FL) helps address this problem.**

**In recent years, Reinforcement Learning(RL) has gained a lot of attention. It has proven its ability in solving exceptionally complex problems that can not be solved using traditional ML algorithms. However, a major problem that RL faces is sample efficiency and data privacy. In that sense, FRL derives its inspiration from the above challenge. FRL is a combination of RL and FL that will enhance the performance of RL while maintaining data privacy using the basic concepts of Federated Learning(FL).**

## Literature Review

Mobile edge caching is an emerging technology for supporting Internet of Things (IoT) services and applications that require massive access to content in a mobile network. Currently, most optimization-based methods do not have the capability to self-adapt in dynamic environments. Xiaofei Wang et al. [1] proposed a federated deep reinforcement learning cooperative edge caching framework (FADE).

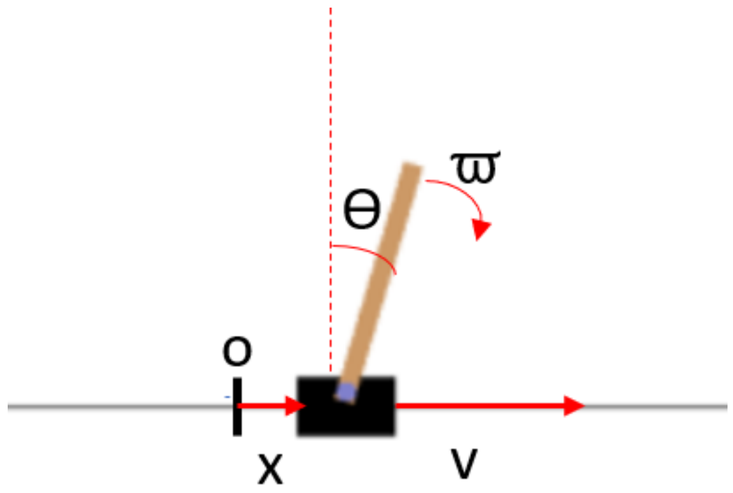
With FADE, IoT devices can learn a shared model while each device keeps its training data thus preserving privacy. FADE is executed in a decentralized manner. First, the IoT device downloads the shared model from the server and trains on it using the local data on the device. Second, the updated local model from the device is sent to the local server, where all the local models from all participants are aggregated to update the global model. [1] model the content replacement problem as an Markov Decision Process (MDP) process. They use a double deep Q-network (DDQN) as their RL method. [1] evaluated FADE in terms of network performance. They compared FADE with some state-of-the-art caching schemes:

1) Least recently used (LRU)
2) Least frequently used (LFU)
3) First in, first out (FIFO)
4) Oracle [2]
5) Centralized DRL [3].

The results show that the FADE framework outperforms the other schemes in terms of average delay, the hit rate, and the traffic offload of backhaul while achieving the approximate performance of the centralized Deep Reinforcement Learning (DRL) scheme with a low loss. Finally, Xiaofei Wang et al. [1] went into detail to explain their proposed framework.

## Environment

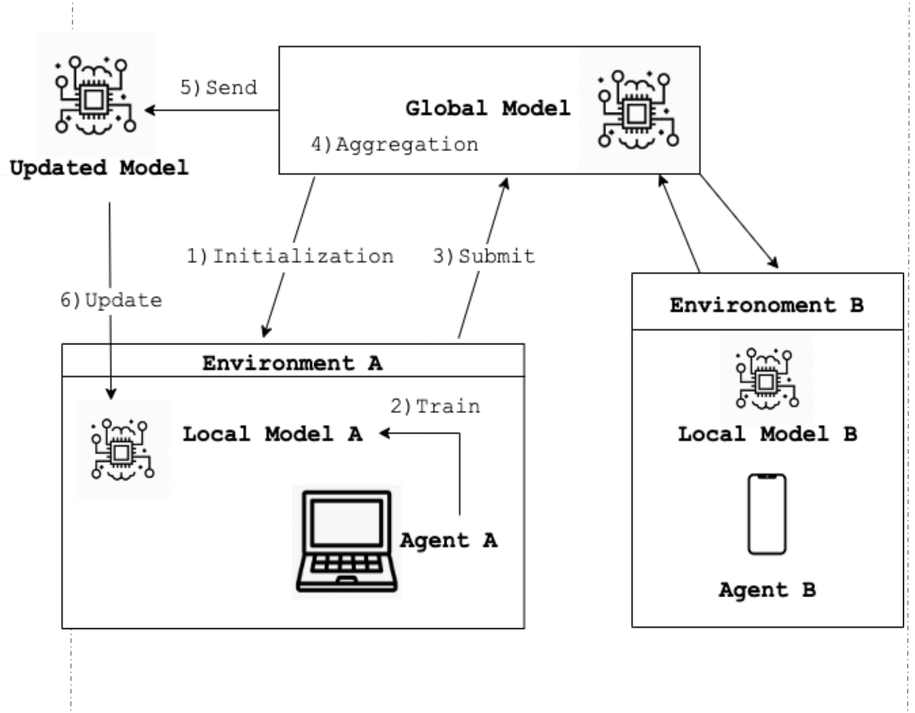
The environment used in this research is the CartPole- v0 environment. CartPole-v0 is one of the Classic Control environments in OpenAI Gym. The goal of CartPole is to balance a pole on a cart by going left or right.




## Methodology

In this research, we used REINFORCE as our RL algorithm. REINFORCE is a Policy Gradient algorithm, which is a type of Policy-Based method. To implement this algorithm, we create a policy, which is a neural network that takes the current state as an input and outputs an action probability distribution or a single action. The policy π tells the agent what action to take given the current state, it is the brain of the agent. We iterate on the policy, adjusting it until we find a policy that solves the problem.

In our FRL architecture , a client-server model is applied. Where the server is responsible for communication with clients and the aggregation of the models sent from the clients. The clients send the server their trained models and the server applies FedAvg [4] on them to obtain a new combined and global model.

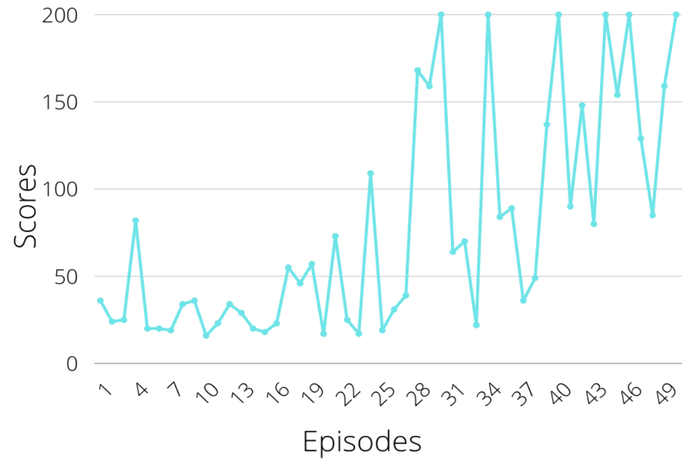



## Results

**A**

Two clients were trained for 50 episodes and FedAvg[4] was performed on them to obtain a new combined model.

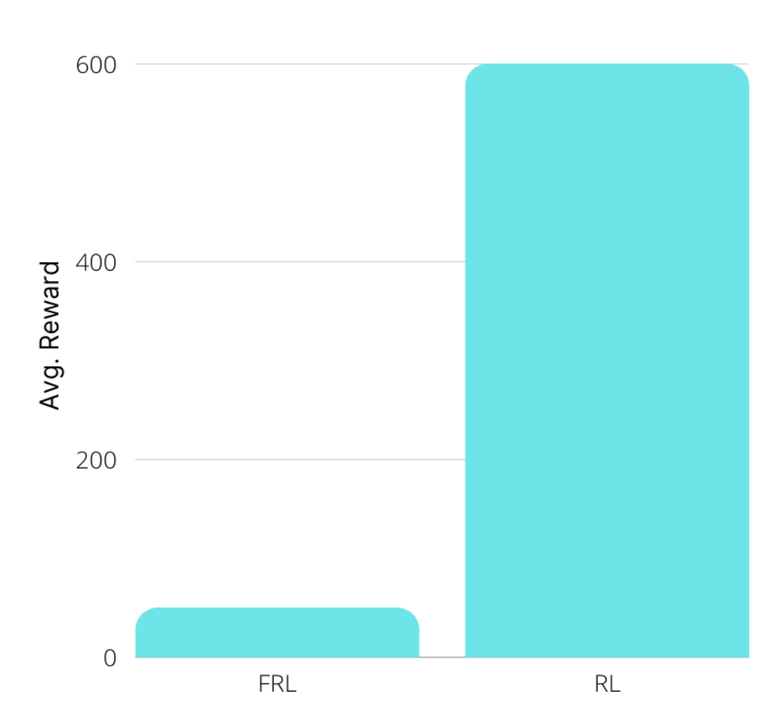
FRL with REINFORCE Agent




Ploting of the FRL agent episodes and scores.

**B**

Our FRL model was compared to a traditional REINFORCE model to observe difference in the learning speed of each.




The FRL agent was able to reach an average reward of 100 in 50 episodes ,while the RL agent was able to reach an average reward of 100 in 600 episodes.

## Conclusion

Applying Federated Learning to Reinforcement Learning applications will not only help preserve data privacy, but will significantly improve the agents learning speed.

## References

1. Wang, X., Wang, C., Li, X., Leung, V. C., & Taleb, T. (2020). Federated deep reinforcement learning for Internet of Things with decentralized cooperative edge caching. *IEEE Internet of Things Journal*, *7*(10), 9441-9455.
2. S. Mu¨ller, O. Atan, et al., “Context-aware proactive content caching with service differentiation in wireless networks,” *IEEE Trans. Wireless Commun.*, vol. 16, no. 2, pp. 1024-1036, Feb. 2017.
3. H. V. Hasselt, A. Guez, and D. Silver, “Deep reinforcement learning with double Q-learning,” in *Proc. AAAI*, pp. 2094-2100, Feb. 12-17, Phoenix, Arizona, USA, 2016.
4. Li, Xiang, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. "On the convergence of fedavg on non-iid data." *arXiv preprint arXiv:1907.02189* (2019).

German University in Cairo

**Prepared for Thesis Poster Display Conference**

11th -12th June 2022

Faculty of Management Technology

THESIS Poster Display Conference 2022