MI Theory Project Synopsis and Literature Survey - H Section, 2022

Project Title:
Time-Optimized Federated Reinforcement Learning

Project Team Details:

Suraj Rao	PES1UG20CS449
Sriram Radhakrishna	PES1UG20CS435
Swapnil Nair	PES1UG20CS452

Abstract

Here's the problem we are solving.

Federated Machine Learning is heavily dependent on communicating with its constituent nodes however networks are unpredictable and dynamic while current algorithms are not, resulting in increased time per iteration , especially in situations where nodes are distant.

Here's what we are doing.

In this project, we aim to develop a new architecture for federation in an IoT network to distribute the agents and their search spaces among the nodes such that the total training time is minimized for a reinforcement learning model.

Here's why it is necessary.

Reinforcement Learning is not sample efficient and in a distributed implementation, the number of iterations and as a direct result, the time taken for every iteration is of utmost importance since a 10% improvement can make a world of difference on a system that endures varying network link speeds and reliability and takes days to train.

Scope

What is inside the scope ?

- **Implementation of a reinforcement learning network**
- **Implementation of the same network on different devices with different sections of the same dataset**
- **Federation of the model across the connected devices in a time-efficient manner by modelling the network as a minimum spanning tree.**

What is outside the scope ?

- **Improving sample efficiency**
- **Implementing the system for a specific use case**

Novelty of the project opted.

- **The implementation of a minimum spanning tree to aggregate the results of the RL models from each node.**

Dataset

Dataset source ?

Dataset size ?

Any preprocessing done, mention ?

Any data analytics used ? why ?

Any statistical analysis used ? Why ?

Machine Learning related Details

Machine learning algorithm - Deep Q Network

ML Library used - Tensorflow

Other python libraries used - matplotlib, numpy, pillow

Model performance metrics - Percentage average improvement from Standard (PAIS)

Design Approach / Methodology / Planning of work

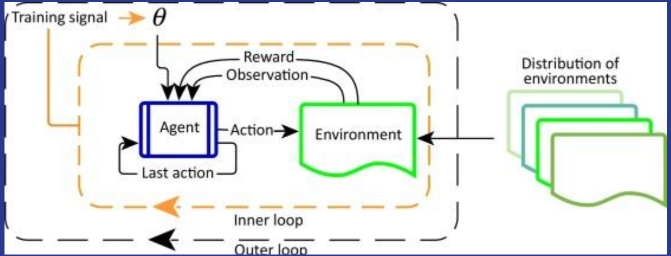
Methodology will include the steps to be followed to achieve the objective of the project during the project development.

We propose doing this by modeling the network as a complete weighted directed graph and generating a minimum spanning tree from the graph by taking the ping between all pairs of devices as the weights for the edges. This concept has use cases in many different scenarios, especially in situations involving robot swarms for simultaneous localization and mapping.

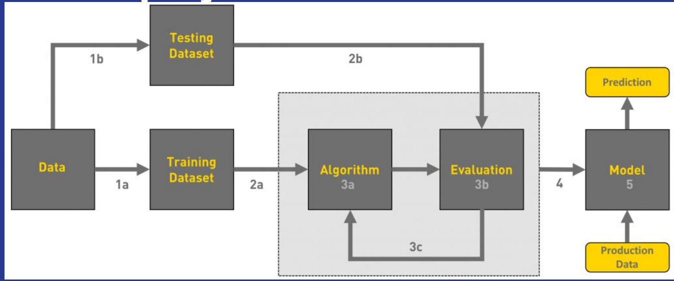
The challenge here will lie in tuning this architecture to produce an aggregated Q-Table such that the RL model operates at a high level of accuracy.

Design Details

High level architecture diagram / Model Architecture



Work flow of the project



Literature Survey

Paper 1

Federated Reinforcement Learning: Techniques, Applications, and Open Challenges

Year of Publication: 2021

Authors: Jiaju Qi, Qihao Zhou, Lei Lei, Kan Zheng

Paper 1

Abstract & Methodology

This paper presents a comprehensive survey of Federated Reinforcement Learning (FRL), an emerging and promising field in Reinforcement Learning (RL).

Starting with a tutorial of Federated Learning (FL) and RL, the paper then focus on the introduction of FRL as a new method with great potential by leveraging the basic idea of FL to improve the performance of RL while preserving data-privacy. According to the distribution characteristics of the agents in the framework, FRL algorithms can be divided into two categories, i.e., Horizontal Federated Reinforcement Learning (HFRL) and Vertical Federated Reinforcement Learning (VFRL). This paper provides the detailed definitions of each category by formulas, investigate the evolution of FRL from a technical perspective, and highlight its advantages over previous RL algorithms. In addition, the existing works on FRL are summarized by application fields, including edge computing, communication, control optimization, and attack detection. Finally, the paper describes and discusses several key research directions that are crucial to solving the open problems within FRL.

Paper 1

Merits & Demerits

Merits:	Demerits/Issues:
• Learning experience can be shared among decentralized multiple parties while ensuring privacy and scalability without requiring direct data offloading to servers or third parties • Learning speed can be greatly accelerated • Aggregate information from both simulated and real world environments, and thus bridge the gap between them	• Learning convergence in HFRL • Agents without rewards in VFRL • Communications • Privacy and Security-data poisoning • Join and exit mechanisms design • Incentive mechanisms • Peer-to-peer cooperation

Paper 1

Conclusion & Motivation

FRL serves an increasingly important role as an enabler of various applications like:

- **Edge computing**
- **Communication networks**
- **Control optimization**
- **Attack detection**
- **Energy management of multiple smart homes**

Paper 2

Optimizing Federated Learning on Non-IID Data with Reinforcement Learning

Year of Publication: 2020

Authors: Hao Wang, Zakhary Kaplan, Di Niu and Baochun Li

Dataset: MNIST, FashionMNIST, CIFAR-10

Paper 2

Abstract & Methodology

Due to the limited network connectivity of mobile devices, it is not practical for federated learning to perform model updates and aggregation on all participating devices in parallel. Besides, data samples across all devices are usually not independent and identically distributed (IID), posing additional challenges to the convergence and speed of federated learning.

This paper proposes FAVOR, an experience-driven control framework that intelligently chooses the client devices to participate in each round of federated learning to counterbalance the bias introduced by non-IID data and to speed up convergence.

Paper 2

Merits & Demerits

Merits:	Demerits:
• FAVOR is the first work that counterbalances the bias from different non-IID data by dynamically constructing the subset of participating devices with DRL techniques. • Faster convergence speed. • Speedup FL training by minimizing the number of communication rounds.	• Demerits have not been looked into in the paper

Paper 2

Conclusion & Motivation

According to the authors of this paper non-IID data exacerbates the divergence of model weights on participating devices, and increases the number of communication rounds of federated learning by a substantial margin. With both mathematical demonstrations and empirical studies, the authors found the implicit connection between model weights and the distribution of data that the model is trained on. Hence, they proposed to actively select a specific subset of devices to participate in training at each round, in order to counterbalance the bias introduced by non-IID data on each device and to speedup FL training by minimizing the number of communication rounds.

An extensive comparison with FL training jobs by FEDAVG(Federated Averaging Algorithm) has shown that FL training with FAVOR has reduced the number of communication rounds by up to 49% on the MNIST dataset, up to 23% on FashionMNIST, and up to 42% on CIFAR-10.

Paper 3

Federated Reinforcement Learning For Fast Personalization

Published at the 2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)

Authors : Chetan Nadiger, Anil Kumar, Sherine Abdelhak

Paper 3

Abstract & Methodology

Personalization has been sought after in gaming, personal assistants, dialogue managers, and other popular application categories. One of the challenges of personalization methods is the time they take to adapt to the user behavior or reactions. This sometimes is detrimental to user experience. The contribution of this work is twofold: (1) showing the applicability of granular (per user) personalization through the use of reinforcement learning, and (2) proposing a novel mitigation strategy to decrease the personalization time, through federated learning.

The current paper propose a grouping policy where the agents are clustered based on similarity criteria for faster learning instead of the more commonly found multiparameter approach. The key novel contribution of this paper is the use of Deep Q Network (DQN) [7] reinforcement learning algorithm in a federated setting to achieve faster personalization. One global model is built in server for each group of similar players (clients). Depending on the application needs, there could be different ways in which players’ similarity could be calculated.

Paper 3

Merits & Demerits

While the merits and demerits haven't been explicitly given a once over in the paper, the general demerits of FRL including the additional overhead and costs associated with implementing it do come into picture. The benefits of having better adaptation and lower learning times is very commonplace and need not be further elaborated.

The table alongside depicts the much higher level of personalization when the number of clients is increased in the same number of iterations or cycles.

Concept	Clients	C1	C2	C3	C4	C5
Inference		12%	15%	20%	16%	15%
RL		19%	18%	19%	18%	18%
FRL	3	22%	18%	18%	NA	NA
FRL	4	24%	21%	26%	26%	NA
FRL	5	21%	20%	21%	28%	30%

Paper 3

Conclusion & Motivation

In conclusion, this paper presents an approach using federated reinforcement learning to speed up the personalization process of smart agents to their environment. The paper presented a grouping policy, a learning policy, and a federation policy which constitute the overall FRL architecture. The results show the efficacy of the approach by testing it on 3, 4, and 5 agents, where the speed up of the personalization time is ~ 17%. This approach is paving the road for advancements in the area of federated continuous learning for the aim of instant personalization to changing environments.

This paper goes on to demonstrate why FRL is an excellent candidate for scenarios where fast adaptation and user privacy are of primary concern.

Paper 4

Lifelong Federated Reinforcement Learning: A Learning Architecture for Navigation in Cloud Robotic Systems

Year of Publication: 2019

Authors : Boyi Liu , Lujia Wang , and Ming Liu

Dataset : Turtlebot environment

Paper 4

Abstract & Methodology

Abstract—This letter was motivated by the problem of how to make robots fuse and transfer their experience so that they can

effectively use prior knowledge and quickly adapt to new environments. To address the problem, we present a learning architecture

for navigation in cloud robotic systems: Lifelong Federated Reinforcement Learning (LFRL). In the letter, we propose a knowledge

fusion algorithm for upgrading a shared model deployed on the cloud. Then, effective transfer learning methods in LFRL are introduced. LFRL is consistent with human cognitive science and fits well in cloud robotic systems. Experiments show that LFRL greatly improves the efficiency of reinforcement learning for robot navigation. The cloud robotic system deployment also shows that LFRL is capable of fusing prior knowledge. In addition, we release a cloud robotic navigation-learning website to provide the service based on LFRL: www.shared-robotics.com.

Paper 4

Merits & Demerits

To show the performance of LFRL, it was tested and compared with generic methods in the four environments.

The shared model was initialized and evolved after training in Env-1. In the cloud robotic system, the robot downloaded the shared model 1G. Then, the robot performed reinforcement learning based on the shared model. The robot got a private model after training and it would be uploaded to the cloud server. The cloud server fused the private model and the shared model 1G to obstain the shared model 2G. With the same mode, follow-up evolutions would be performed. The authors constructed four environments, so the shared model upgraded to 4G. In Env2-Env4, LFRL increased accuracy of navigating decision and reduced training time in the cloud robotic system. It was observed that the improvements are more efficient with the shared model. LFRL is highly effective for learning a policy over all considered obstacles. It improves the generalization capability of our trained model across commonly encountered environments. Experiments demonstrate that LFRL is capable of reducing training time without sacrificing accuracy of navigating decision in cloud robotic systems.

Paper 4

Conclusion & Motivation

The paper presented a learning architecture LFRL for navigation in cloud robotic systems. The architecture is able to make navigation-learning robots effectively use prior knowledge and quickly adapt to new environment. Additionally, it presented a knowledge fusion algorithm in LFRL and introduced transfer methods. The approach is able to fuse models and asynchronously evolve the shared model. The architecture was validated and algorithms in policy-learning experiments and released a website to provide the service.

The architecture has fixed requirements for the dimensions of input sensor signal and the dimensions of action.

The paper doesn't cover the future work necessary to make LFRL flexible to deal with different input and output dimensions. The more flexible LFRL will offer a wider range of services in cloud robotic systems.

Paper 5

Federated Transfer Reinforcement Learning for Autonomous Driving

Year of Publication: 2019

Authors : Xinle Liang, Yang Liu, Tianjian Chen, Ming Liu, Qiang Yang

Dataset: Airsim coastline map

Paper 5

Abstract & Methodology

Abstract— Reinforcement learning (RL) is widely used in autonomous driving tasks and training RL models typically involves in a multi-step process: pre-training RL models on simulators, uploading the pre-trained model to real-life robots, and fine-tuning the weight parameters on robot vehicles. This sequential process is extremely time-consuming and more importantly, knowledge from the fine-tuned model stays local and can not be re-used or leveraged collaboratively. To tackle this problem, we present an online federated RL transfer process for real-time knowledge extraction where all the participant agents make corresponding actions with the knowledge learned by others, even when they are acting in very different environments.

To validate the effectiveness of the proposed approach, we constructed a real-life collision avoidance system with Microsoft Airsim simulator and NVIDIA JetsonTX2 car agents, which cooperatively learn from scratch to avoid collisions in indoor environment with obstacle objects. We demonstrate that with the proposed framework, the simulator car agents can transfer knowledge to the RC cars in real-time, with 27% increase in the average distance with obstacles and 42% decrease in the collision counts.

Paper 5

Merits & Demerits

	car1		car2		car3	
	avg_dist	coll_no	avg_dist	coll_no	avg_dist	coll_no
DDPG	0.39	18	0.29	31	0.38	24
FTRL-DDPG	0.42 (7.7%)	**9** (50%)	0.37 (27.6%)	27 (12.9%)	**0.51** (34.2%)	17 (29.2%)
FTRL-DDPG-SIM	**0.45** (15.4%)	12 (33.3%)	**0.39** (34.5%)	**16** (48.4%)	0.50 (31.6%)	**13** (45.8%)

As a conclusion, for the autonomous driving areas, with the capabilities of transferring online knowledge from simulators to real-life cars, FTRL-DDPG-SIM performs better than both single execution of single RL agents and federation model with identical RL agents with better training speed and performance.

Paper 5

Conclusion & Motivation

In this work, the FTRL framework was presented, which is capable of conducting online transfer to the knowledge of different RL tasks executed in non-identical environments. However, the transfer model employed in FTRL presented in this work is rather simple, which is based on human knowledge. Autonomously transferring the experience or knowledge from the already learned tasks to new ones online constitutes another research frontier.

THE END