The City College of New York

# CSC 36000:
# Modern Distributed Computing *with AI Agents*

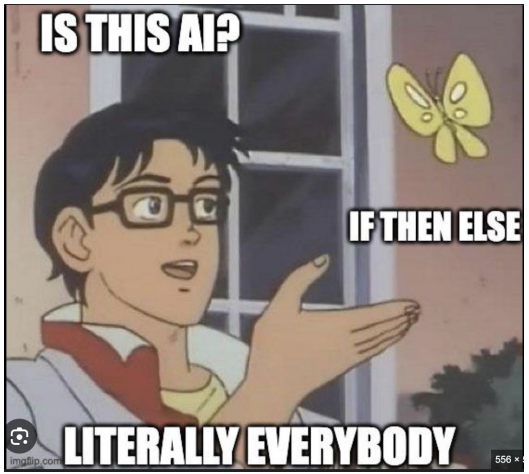
By Saptarashmi Bandyopadhyay
Email: sbandyopadhyay@ccny.cuny.edu
Assistant Professor of Computer Science
City College of New York and Graduate Center at City University of New York

# About Me

- PhD in Computer Science from the University of Maryland, College Park in Summer 2025
- Recent PhD Student Researcher at Google AI AR and Google DeepMind and as an AI Resident at Google X
- Lead PhD Researcher in a DoD LTS project
- Tenure-Track Assistant Professor of Computer Science at the City College of New York beginning Fall 2025

# Research Interests

- AI Agents
  - Improving capabilities of AI Agents
    - Real World Applications in Climate Conservation, Supply Chain Orchestration, Multimodal Agents, LLM Agents, Recommender Systems, Economics, AI Privacy etc
- Distributed Training (we know it as Modern Distributed Computing)
- Multimodal Deep Learning
  - Interpretable Semantic Understanding across Image, Text, Video, Audio modes
- AI Alignment
  - Human-AI collaboration; Fixing Mistakes of Humans/other AI Agents; Explainable AI
- AI Agents Seminar Series: (go.umd.edu/marl)



# About You

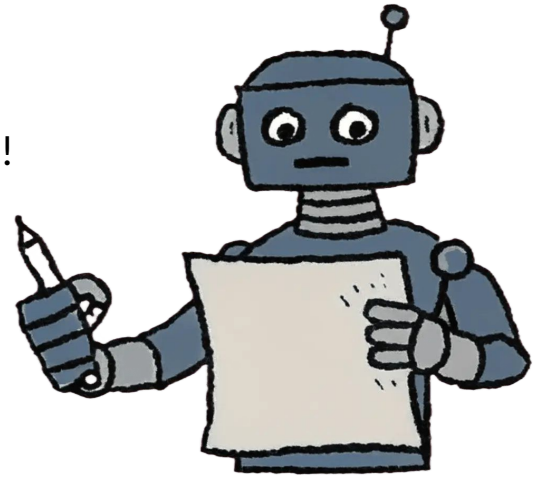
- What programming background do you have?
- What are you most excited to learn?
- What do you dream to do after graduation?
- What is your expectations from this course?

# About this Course

- Website: https://saptab.github.io/modern-distributed-computing-with-AI-Agents/
- Programming Language: Python
- We will be learning about distributed computing architectures with a focus on real-world applications like AI Agents
    - Federated Learning
    - Decentralized Learning
    - Parallel Processing
    - LLM/VLM Agents
- At the end of the semester you will be able to understand these systems and implement them in a real-world context

# Grading Structure

- Subject to changes to be confirmed by September 3!
- Four programming assignments (20%)
- Four written homework assignments (20%)
- Midterm Exam (15%)
- Group Project (25%)
- Final Exam (20%)



# How to Reach Me

- Email: sbandyopadhyay@ccny.cuny.edu
- Office: NAC 7/244 (Being set up by CUNY Facilities right now)
- Office Hours: TBD

# This Lecture

- Motivation of Distributed Computing
- Overview of Machine Learning, AI Agents, and their applications

# Why Distributed Computing?

# Why not use one GIANT computer?

- Every second, Google handles 100,000+ searches
- Netflix is streaming video to 270 million subscribers all over the world
- Using a single computer to this is IMPOSSIBLE

# Physics and Money

- Putting that many transistors on a chip would literally cause them to melt
- The speed of light introduced delays that can’t be overcome, no matter how clever we are
- It would cost way too much!

# AI Agents: Overview and Applications

# What is Machine Learning?

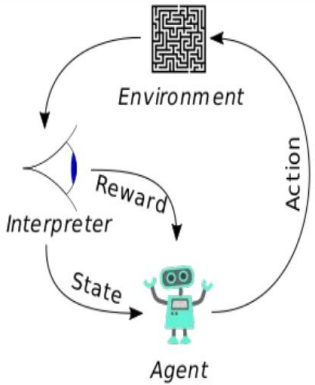
- Algorithms that help to automatically perform tasks
- Fundamental concepts involve High School Mathematics
  - y = f(x)
- Data may be texts, images, videos, audios etc. So embeddings are used to convert them to numbers for numerical operations

# Multi-Agent Reinforcement Learning

- Evolving area of Machine Learning (specially since 2018)
- Multiple autonomous agents can interact with each other to foster competition and cooperation among humans and autonomous agents
- These agents try to maximize their individual interests while optimizing the greater good.
- E.g. 2 or more self-driving cars would try to reach their respective destinations at the earliest but would want to stop allowing the other car to pass in order to avoid accidents.

# Single-Agent vs. Multi-agent RL (DeepMind, 2019)

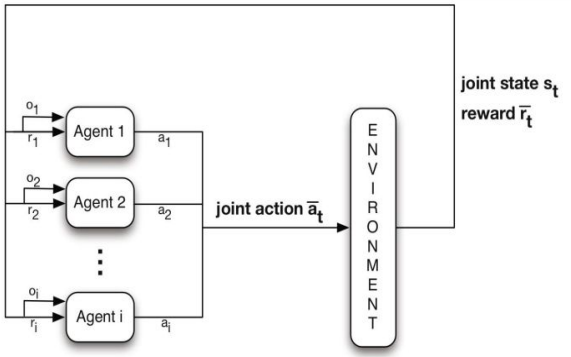
## Traditional (Single-Agent) RL




Source: Wikipedia



## Multiagent Reinforcement Learning




Source: Nowe, Vrancx & De Hauwere 2012

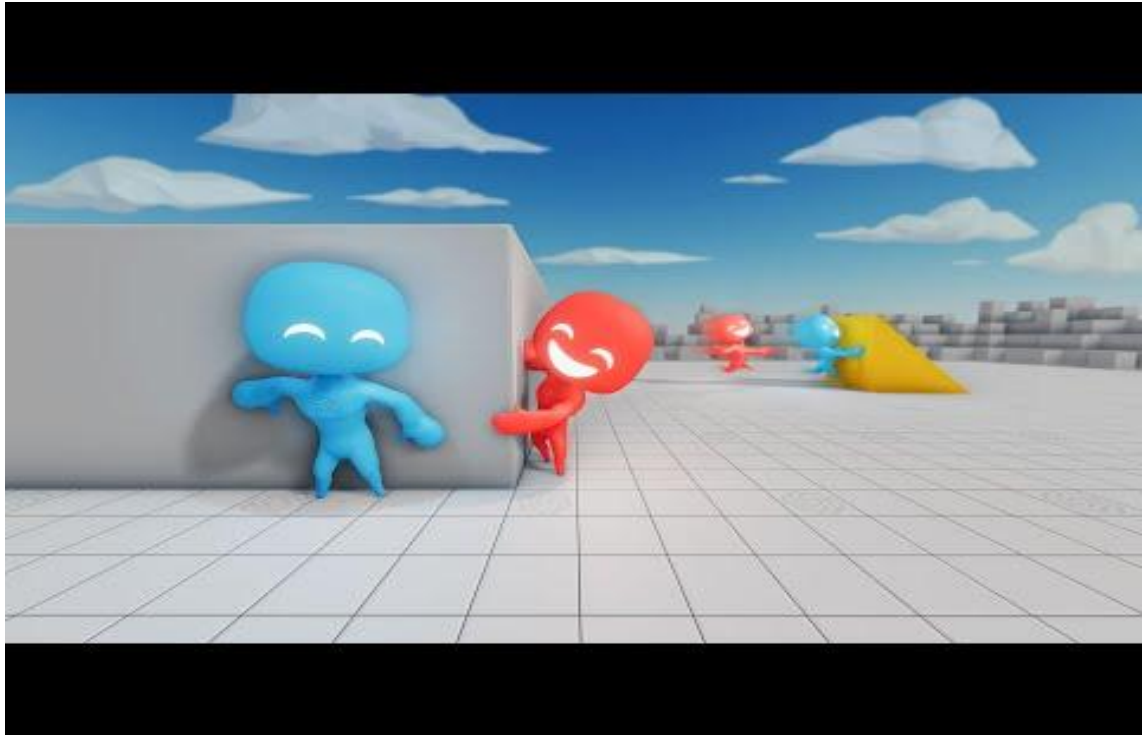
# Motivation from Game Theory

- Studying strategic interaction of rational agents to achieve a goal(s)
- Stackelberg games: Attackers try to gain control of secure targets protected by defenders
- Interdiction games: Extension of Stackelberg security games where attacks are constrained by a path on a graph interdiction environment G = (V,E) where V=#vertices; E =#edges
- $V = S \cup T$ where S is a set of vertices for sources and T is a set of vertices for targets
- Nash Equilibrium: Each player knows the equilibrium strategies of the other player. Players have no incentives to deviate from their strategies unilaterally
- Multi-agent Reinforcement Learning research is inspired by Game Theory work

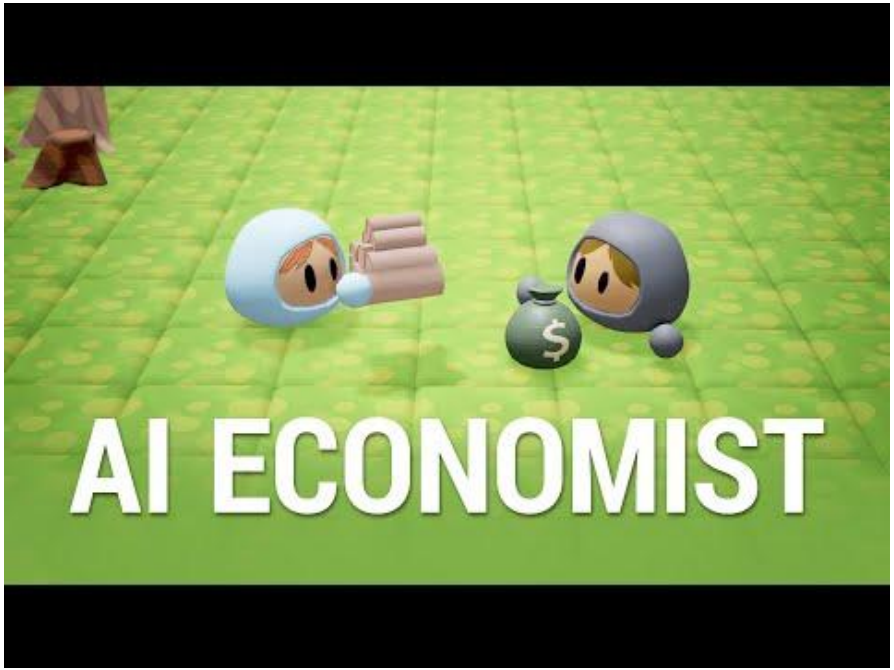
# Some Applications

- Game theory problems
- Discovering, analyzing and disrupting illicit networks
  - Arms
  - Humans
  - Wildlife
  - Deforested trees
  - Drugs
  - Counterfeit goods
- War games and strategies (nuclear arms race, world wars)
- Equitable markets
- International relations like peace treaty negotiations
- Robust and Fair Management/Governance
- Fair tax policies
- Climate change
- Existential disasters like pandemics
- Swarm robotics
- Internet of Things (IoT)

# Multi-agent RL for Hide-and-Seek (OpenAI, 2019)



# Tax Policy with Multi-agent RL (Zheng et. al, 2020)



# Data Science

If the only tool you have is a relational database, everything looks like a table.

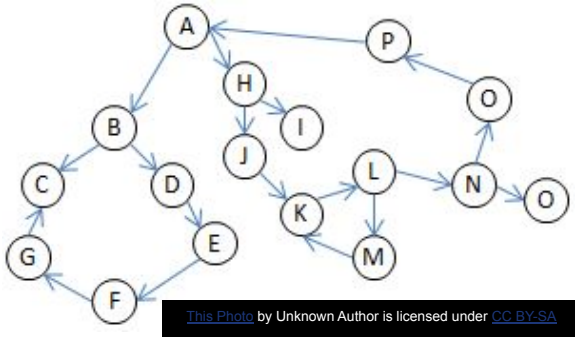
- Data management is important with big data having different modalities like texts, images, videos
- **Relational Databases** are used prevalently to store data in tables with many attributes
- Data Analysis is critical to detect patterns that are useful to AI

# Example: Graph based Data in Social Media

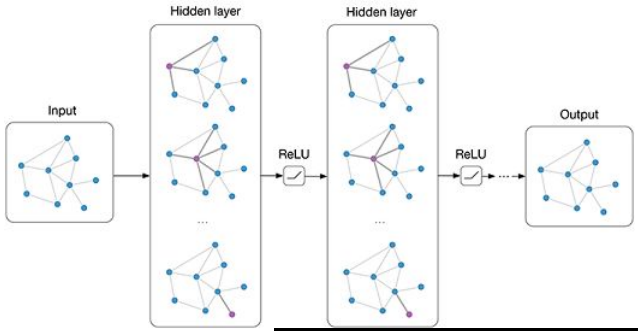
- A graph is denoted as G(V,E) where V is the set of vertices and E is the set of edges connecting vertices in V.

- Social Networks involve users interactively engaging with other users as nodes and their interactions are edges.

- Learning paradigm also changes with graph based learning in many situations

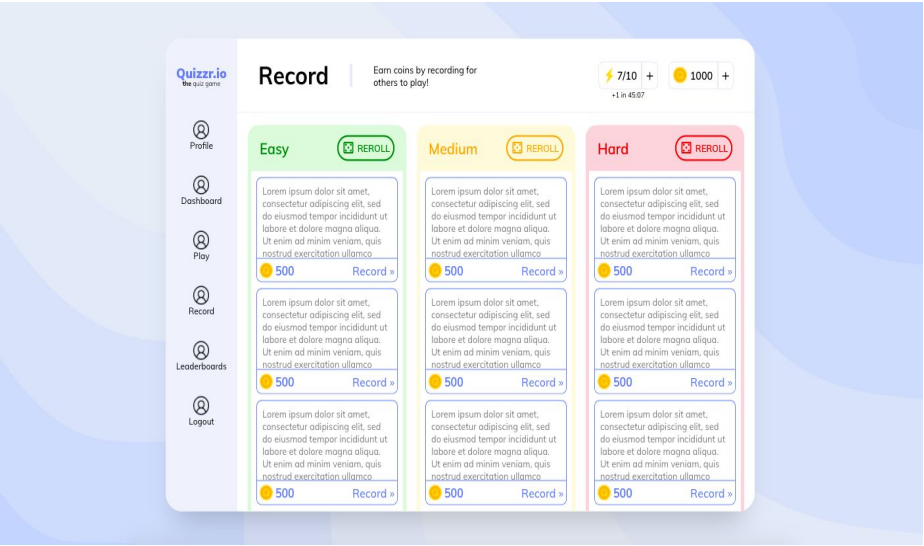




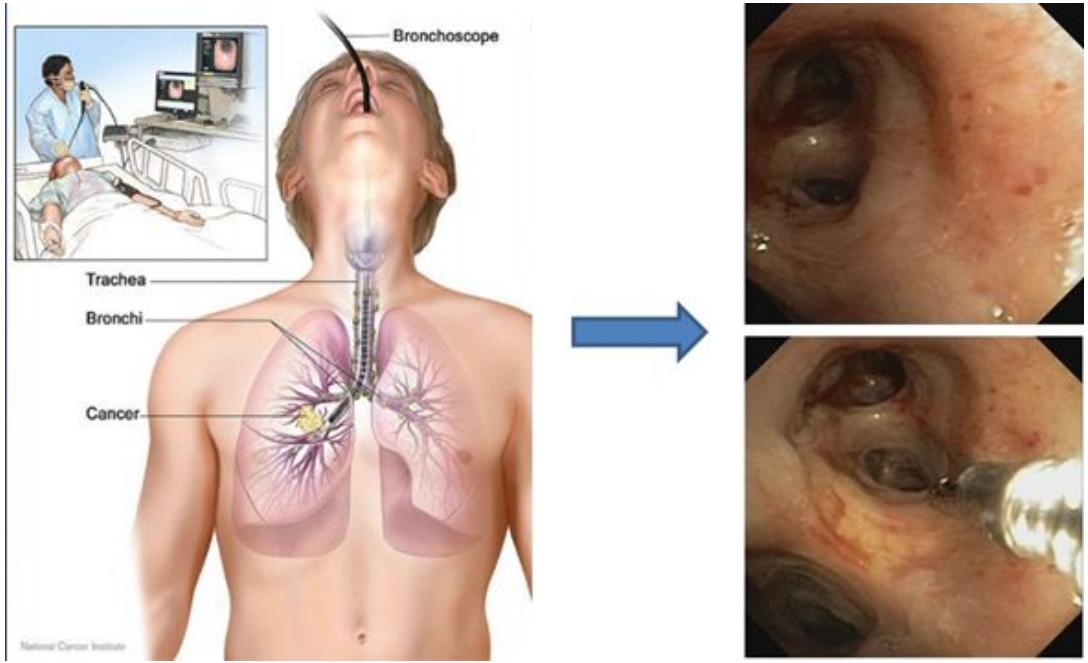

Src: Applications of GNN's (click on image)

# Human Computer Interaction



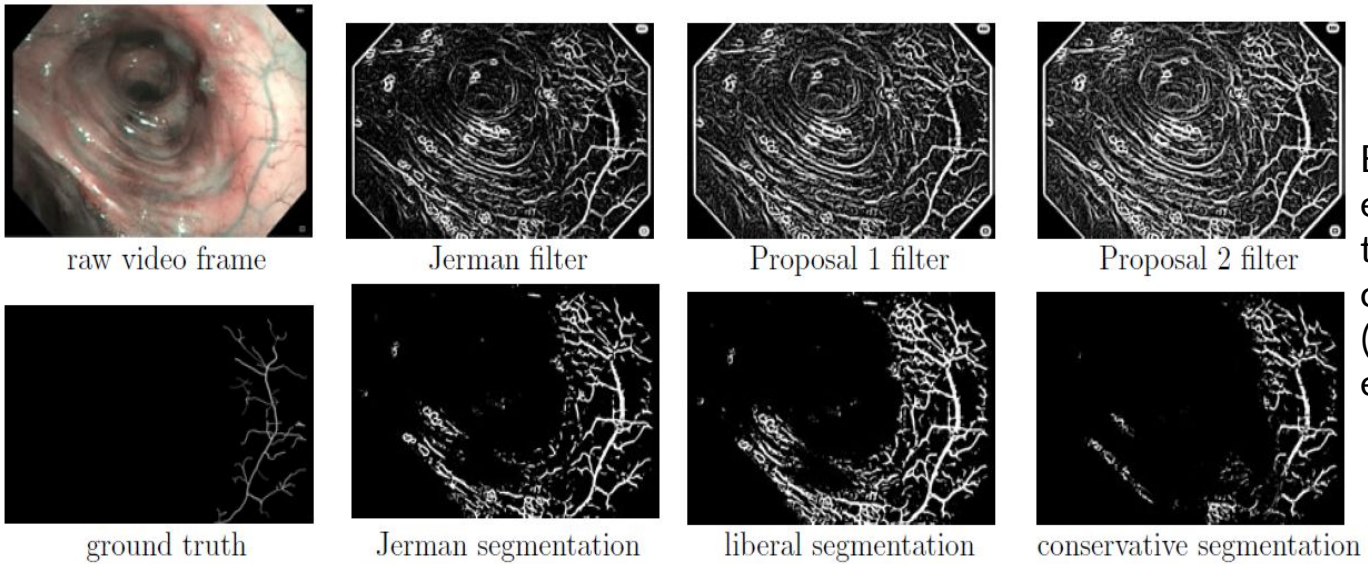

- Website interfaces are developed for user interaction, recommendations with AI & Data Collection

# Medical Imaging & Computer Vision




- Endoscopy: Visualization of airway
- Useful for early lung cancer detection and treatment

# Medical Imaging & Computer Vision



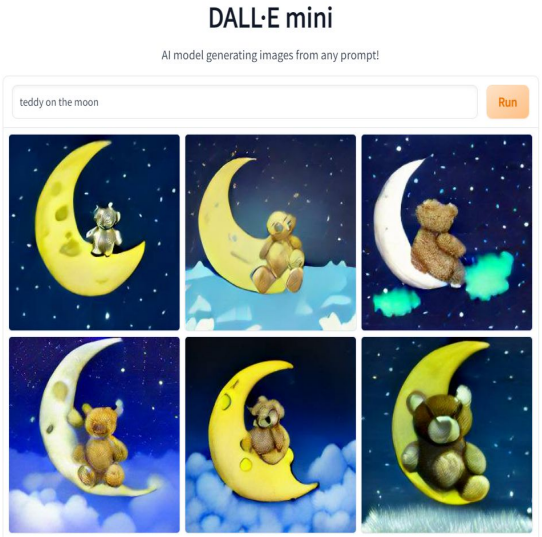

Blood vessel extraction from the airways of a cancer patient (Bandyopadhyay, et al, 2020)

# Natural Language Processing

1. Machine Translation - Google Translate/Siri
    a. Multilinguality; Low Resource Languages
2. Question Answering - IBM Watson
3. Summarization
4. Natural Language Generation
5. Fairness of NLP Systems

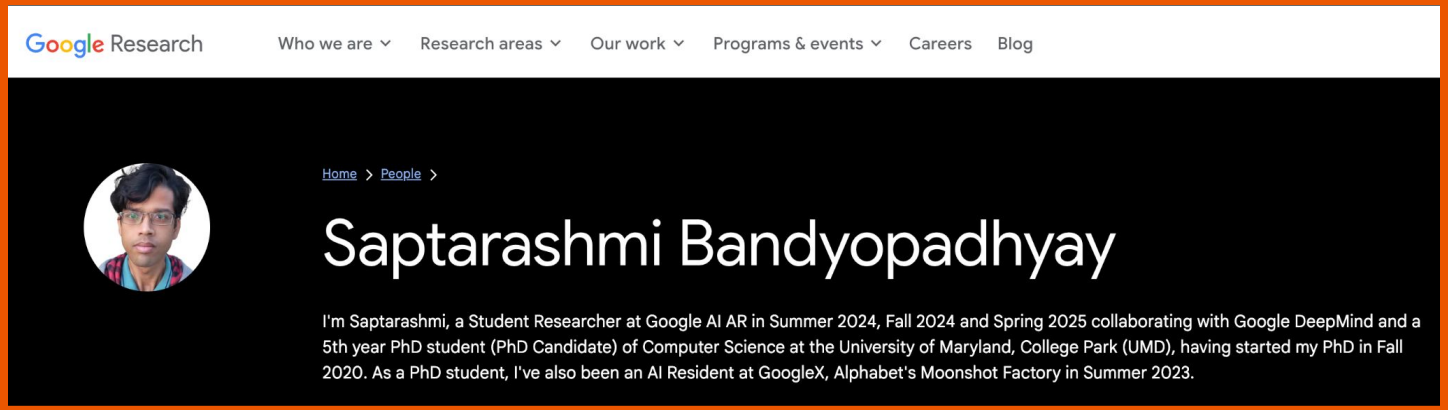
# Multimodal Learning

- Learning from text, images, audios, videos (any mode of information)
- Prominent models like GPT, DALL-E (by OpenAI) https://huggingface.co/spaces/dalle-mini/dalle-mini
- Use Hugging Face pretrained models which are open -access



# Case Study:
# Multimodal AI Agents (in AR/VR)

Research Done during my PhD Student Researcher-ship at Google AI AR and Google DeepMind since June 2024
Arxiv PrePrint (under submission)

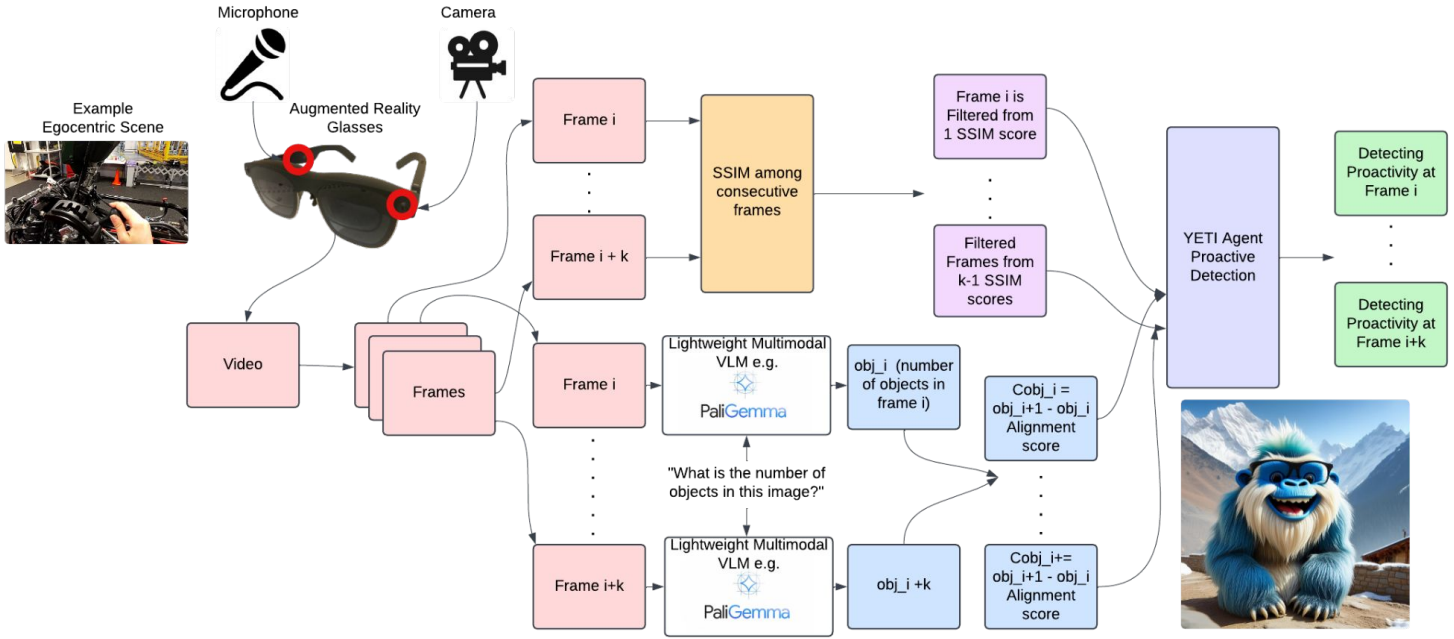


# Research Question for Augmented Reality Agents

- How can AI Agents autonomously assist humans with Augmented Reality (AR)?
  - When should an AI Agent intervene while guiding plans for everyday tasks?
  - How can AI Agents autonomously intervene in our phones or AR devices efficiently?
- YETI is able to efficiently detect when the AI should step in to help the human

| Features | Size (MB) | × SSIM | × CObj |
|---|---|---|---|
| Depth Estimation | 137,408 | 6,543 | 6,870 |
| Eye Gaze (E) | 617 | 29 | 31 |
| Hand Pose (H) | 53,749 | 2,660 | 2,688 |
| Head Pose | 1,141 | 54 | 57 |
| IMU (I) | 1,132 | 54 | 57 |
| **SSIM** (Ours) | **21** | | |
| **Alignment Cobj** (Ours) | **20** | | |

# YETI (YET to Intervene) Multimodal Proactive AI Agent



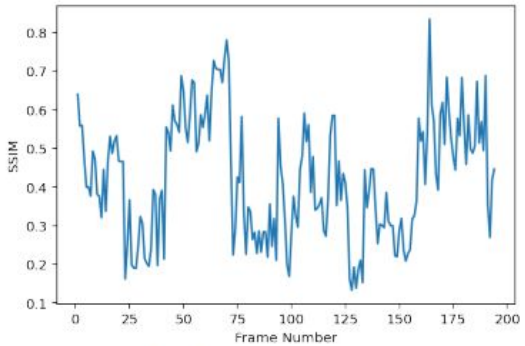
# Demonstration for Proactive Multimodal Agent in AR

(a) Intervention - 3 seconds

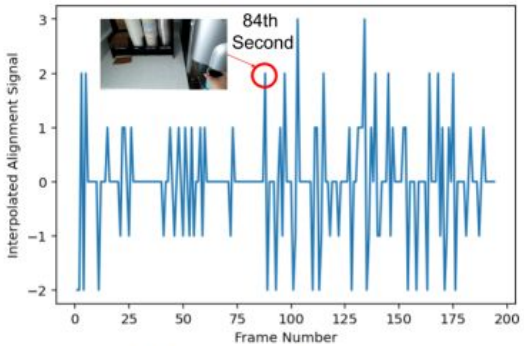
(b) Intervention - 2 seconds

(c) Intervention - 1 second

(d) AI Agent proactively intervenes.




(e) SSIM between consecutive frames.




(f) Changing Objects Alignment Signal

Proactive Multimodal AI Agent guiding how to make coffee with Augmented Reality (AR)

# Multi-Agent Autonomous Orchestration of Global Supply Chains

AI Residency at Google X in Summer 2023
(Open Source Research) Under Submission with Open Source Data

# Multi-agent Decision Making in Global Supply Chains

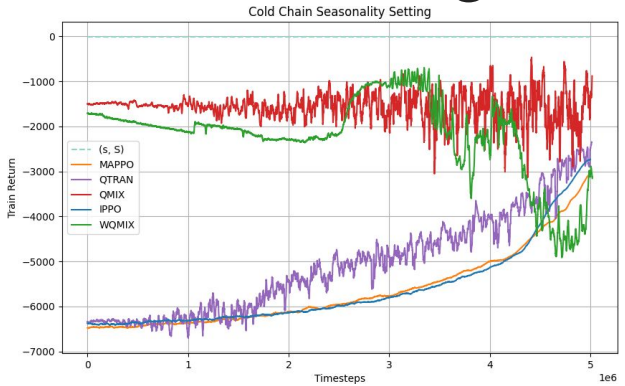
- Global supply chains involving multiple agents enable good movements worth trillions of $s
- There are agents sailing ships carrying good from manufacturing zones to consumer markets
- Then agents drive vehicles or fly from the ports to warehouses to shops or last-mile deliveries
- Agents can also fly goods to remote locations or over difficult terrain
- It is vital to be resilient as seen during the COVID-19 pandemic which shut down supply chains globally

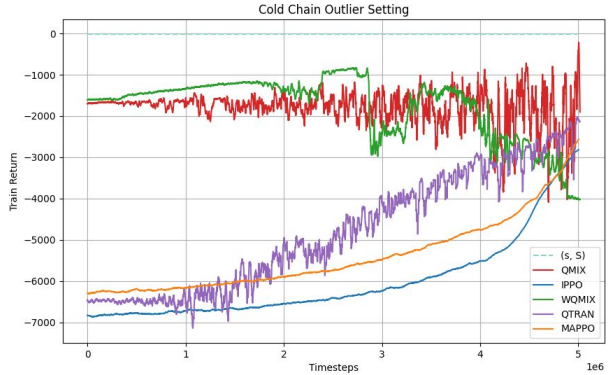
# Orchestration with Multi-agent Reinforcement Learning

- **IPPO**: On-Policy Independent Learning scaling PPO to each Agent individually
  - **Stability challenges** as each agent has to scale up their actor & critic losses
- **MAPPO**: Applying On-Policy PPO to Multiple Agents with a Joint Reward
  - **Stable rewards** (profits) can help tide over supply chain instabilities
- **QMIX**: Factorizes the joint-action value function with an off-policy Multi-Agent RL algorithm with relaxed constraints over monotonic functions
- **QTRAN**: Off-policy Multi-Agent RL algorithm with constrained factorization of the joint action-value function to generalize better in comparison to QMIX
- **Weighted QMIX (WQMIX)** addresses challenges of convergence in QMIX by giving weights to agent's action-value functions to approximate joint Q value
- **Base Stock Constraints** to help address profit margins during market losses in SKUs

# Results for Multi-Agent Supply Chains

Cold-chain SKUs with seasonality




Cold-chain SKUs with outliers

| Algorithm | IPPO | IPPO Base Stock | QTRAN | QTRAN Base Stock | QMIX | Weighted QMIX | MAPPO |
|---|---|---|---|---|---|---|---|
| Mean Profit | 1007 | 38797 | 33601 | 34427 | 45946 | 81089 | 9020 |
| Time Taken | 14h 24m | 14h 51m | 20h 36m | 14h 50m | 36h 22m | 36h 52m | 14h 21m |

Outliers

Seasonality

| Algorithm | IPPO | IPPO Base Stock | QTRAN | QTRAN Base Stock | QMIX | Weighted QMIX | MAPPO |
|---|---|---|---|---|---|---|---|
| Mean Profit | 8940 | 37352 | 38741 | 36937 | 123601 | 85720 | 11055 |
| Time Taken | 14h 45m | 14h 6m | 20h 58m | 9h 12m | 36h 9m | 36h 42m | 14h 32m |

# Multi-Agent Decision Making for Climate Conservation: Learning Vulnerable Deforestation Hotspots

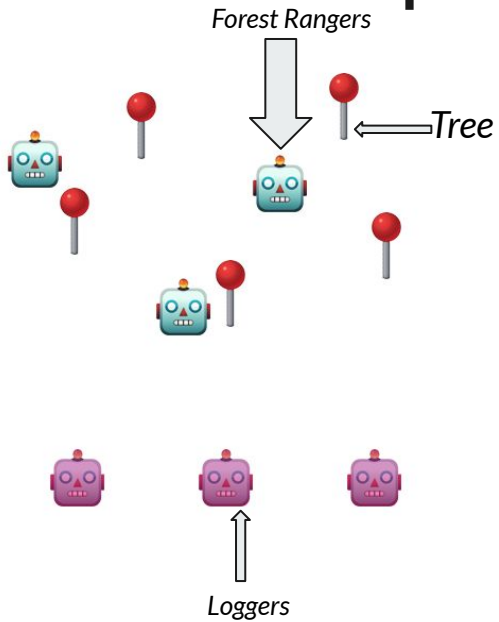
AAAI FSS 2022
AAAI 2022 AI2ASE Workshop

# Motivation for Decision Making to Mitigate Deforestation

- There was minor tree cover loss in Indonesia just 20 years ago in the 2000s
- However deforestation drivers like farming and palm oil production in Indonesia led to rapid loss of tree cover from 2010-2020

# Stackelberg Security Games (SSG) Deforestation Example

- Coordinating drone patrols: Three defender agents (forest rangers) operate cooperative drones to protect five targets (5 regions of vulnerable deforestation regions) from three attackers (loggers)
- The control policies operate at the level of sensor readings and flight-control actions for each individual drone
- Embedded within this high dimensional, continuous problem is the SSG of deciding which targets should be patrolled by the drones
- The goal is to decide how to allocate limited resources (drones) between patrolling targets (vulnerable deforestation regions with valuable trees)
- We don't need to consider the control problem of flying a drone here



# Agents Involved in the Deforestation Environment

- There are defender agents in deforestation like rangers in a national park, protecting trees
- There are attacker agents in deforestation like loggers cutting trees illegally in a dangerously large scale
- Defender agents and attacker agents have competing goals
- Multi-agent decision making is aimed to assist defender agents with policies to secure vulnerable assets (like trees) from attacker agents subject to resource constraints

# Learning Vulnerable Deforestation Hotspots

- Attacks on tree cover, a green security asset, in subnational regions of Indonesia can be accurately predicted with AI (BoostIT Decision Tree)
- Finding vulnerable areas in Green Security Games for strategizing by defenders is a very important problem to solve
- We find that a boosted Decision Tree
  - doesn't use much computing power
  - is accurate in its predictions, and
  - is scalable for managing forest resources efficiently

# Deforestation Vulnerability Prediction Results

| Model | Accuracy | Vulnerable | | Not Vulnerable | |
|---|---|---|---|---|---|
| | | Precision | Recall | Precision | Recall |
| Base model | 62% | 72% | 76% | 27% | 23% |
| Base model with BoostIT | 67% | 79% | 77% | 35% | 32% |
| Base model with terrain features | 69% | 77% | 79% | 51% | 48% |
| Base model with terrain features and BoostIT | 73% | 80% | 83% | 59% | 55% |

Table 1: Performance of different versions of our model on the test data.

# Other emerging areas in Computer Science

- Multilingual Learning (specially Low Resource settings)
- Symbolic Learning and Logical Reasoning
- Explainable and Interpretable Machine Learning
- Adversarial learning (e.g. GANs)
- Quantum Computing (overlaps with AI like Optical Machine Learning)
- Internet of Things (overlaps with AI like Multiagent Learning)

# Questions?