COMPUTATIONAL APPLICATIONS TO POLICY AND STRATEGY (CAPS)

Session 1 – Introduction to CAPS

Leo Klenner

Outline

1. Admin
2. About CAPS
3. Game Theory, Games and AI
4. StarCraft II
5. AI Policy and AI Strategy
6. Required Software

Workshop Information

- Six sessions
- Always Friday 4:00 to 5:00 pm, might run 30min over
- Course website: https://git.io/fAaJn
- Contact
  - Leo (lklenne1@jhu.edu) # general
  - Jonathan (jliu161@jhu.edu) # coding
  - Henry (hfung4@jhu.edu) # data mining

Goals

> Leverage state of the art research on AI gameplay
> Explore how computational methods can advance IR research
> Provide you with a valuable short project and a personal website
> Python is a tool not the primary object of learning itself
> All of the sessions are prepared but we maximize value by working as a team

Goals – Class Project

- Short technical report on any of the topics we discuss or coding exercise
- Examples: memo on constraints of deploying an AI system in a specific scenario, report on StarCraft II as an environment to conduct strategy research, build or manipulate a bot
- We will discuss the project in more detail after session 3
- Find templates on the course website

Core Resources

- Readings and additional learning resources on our GitHub: https://git.io/fAaJn
- Yannakakis and Toeglius. Artificial Intelligence and Games. 2018
  - Readable overview of the field
  - Available online
- Heinold. A Practical Introduction to Python Programming. 2012
  - Comprehensive guide to Python's syntax
  - Available online
- Boddington. Towards a Code of Ethics for Artificial Intelligence. 2017
  - Valuable analysis of the need to bring policy and ethics to AI
  - Available online through the JHU library

Workshop Overview

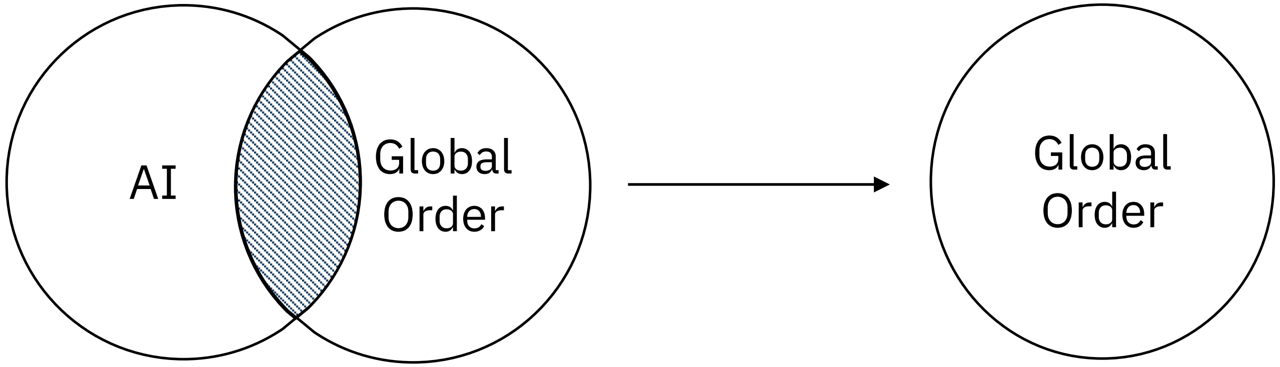
> Session 1: Introduction to CAPS and background on AI and games
> Session 2: Primer on Python
> Session 3: StarCraft II recap and building a rule-based bot
> Session 4: Introduction to basic AI and building a learning-based bot
> Session 5: Game data mining and discussion of AI Policy and Strategy
> Session 6: Creating and hosting a free personal website through GitHub

About CAPS

> How can computational methods advance our understanding and practice of international relations?

> Why is it necessary that the IR community at large engages with emerging technologies such as AI?

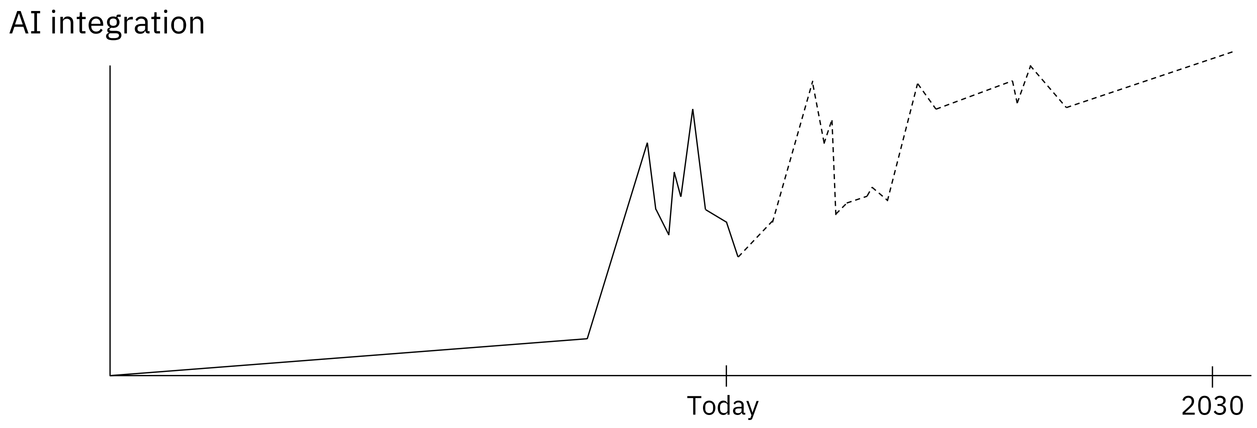
The Intersection of AI and IR



Some keywords:

- Autonomous weapons systems
- Economic disruption
- Human security
- Predictive analysis
- Simulation
- Surveillance

Fluctuations in the Integration of AI



What are the Consequences?

"The Age of Reason originated the thoughts and actions that shaped the contemporary world order. But that order is now in upheaval amid a new, even more sweeping technological revolution whose consequences we have failed to fully reckon with, and whose culmination may be a world relying on machines powered by data and algorithms and ungoverned by ethical or philosophical norms."

- Henry Kissinger, *How the Enlightenment Ends* (June 2018)

How Companies React

"The only way through our "Crisis of Trust" is adopting a set of core values that allows us to navigate these complex times. We are now all stewards of the ethical & humane use of tech."

- Mark Benioff, CEO Salesforce, Tweet (09/10/2018)

"I think one of the only 'arms races' AI people are excited about is the emerging competition between AI research labs to staff up meaningful policy organizations."

- Jack Clark, Policy Lead OpenAI, Tweet (11/10/2018)

Who is in the AI/IR Market?

AI + AI Policy Labs

DeepMind

OpenAI

Venture and Partnerships

DEFENSE INNOVATION UNIT

PARTNERSHIP ON AI

Applications

Palantir

Hala Systems Inc.

PRIMER

AI Policy Research

Future of Humanity Institute
UNIVERSITY OF OXFORD

DARPA

Oxford Digital Diplomacy Research Group

External Stakeholders

UNDP

facebook

salesforce

Reassessing the Consequences

> Do you think Kissinger‘s prediction is right?
> What might be reasons to disagree with it?

Defining AI

> Russel and Norvig (1995)

a. Systems that think like humans
b. Systems that think rationally
c. Systems that act like humans
d. Systems that act rationally

> Obstacles

a. How do humans solve problems?
b. How to formalize uncertain knowledge?
c. How to successfully pass the Turing Test?
d. How to always do the right thing?

The Rational Agent Approach

> Properties

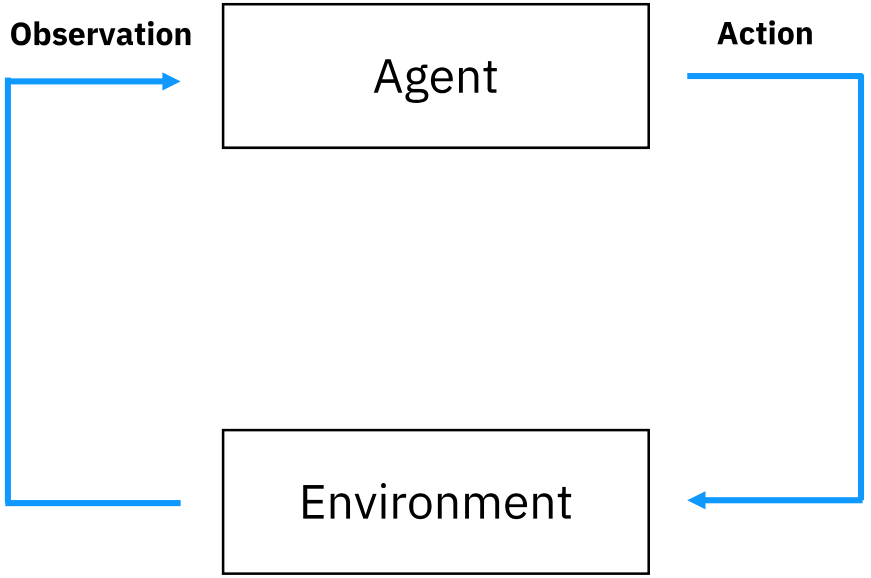
a. Rational agency does not necessarily depend on correct inference
b. Standard for rationality is clearly defined and completely general
c. Limited rationality: can‘t always do the right thing in complicated environments

> Advantages

a. Allows easy comparison between AI and non-AI agents
b. Familiar from game theory and easily relatable to gameplay

Agents Interact with Environments

> General



> Prisoner’s Dilemma

		P2	
		Cooperate	Defect
P1	Cooperate	(3,3)	(0,5)
	Defect	(5,0)	(1,1)

Prisoner’s Dilemma – Agent

Properties of the PD agent	
Behavior	Maximize rewards
Lifetime	1 action
Personality	Homogenous
Memory	No memory
Strategies	Defector, Cooperator, Random, TFT, …

Prisoner’s Dilemma – Environment

Properties of the PD environment	
Type	Game theory, non-zero sum
Gameplay	Turntaking
Action space	2 = Cooperate, Defect
Environment states	4 = CC, CD, DC, DD
Rule of transition between states	Discrete, based on the agents’ actions
Rewards assigned to each state	3/3, 5/0, 0/5, 1/1
Reward horizon	Instantaneous
Mode of information	Perfect Information
Mode of action	Simultaneous

An Interim Conclusion

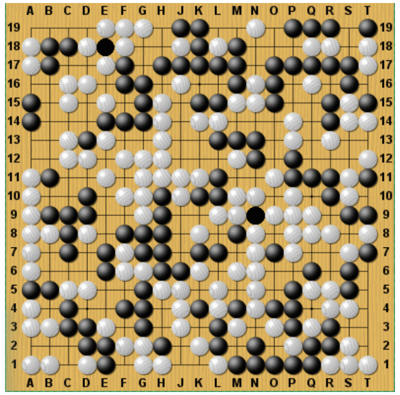
> Intelligent agents are conditioned by their environment

> Consider the environment before you consider the agent

> Games provide powerful, scalable environments

Games and AI

> Two contemporary milestones in AI gameplay



- 1997 Deep Blue beats Garry Kasparov
- Algorithm: Alpha-Beta Search
- Approach: Brute force

- 2016 AlphaGo beats Lee Sedol
- Algorithm: Hybrid (Neural nets, MCTS)
- Approach: Deep learning

From Games to Video Games

“For decades we've been working on games like Chess and Go and the next step is getting closer to the real world because that's where it can really help us.”
- Christie Dennison, ML Engineer OpenAI, speaking at the OpenAI Five Benchmark (5/8/2018)

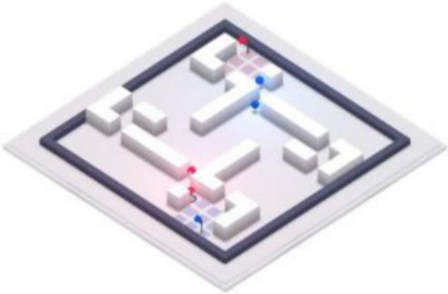
“Throughout the history of AI there has been a focus on strategic games but now AI outfits are pushing for more complexity in multiplayer online battle games and these algorithms and environments are a far cry from the complexity of the real world in the fog of war but then again so are the wargames that we train our commanders in and test our strategies in.”
- Andrew J. Lohn, Engineer RAND Cooperation, speaking at the DIB Public Meeting at SAIS (10/10/2018)

Video Games

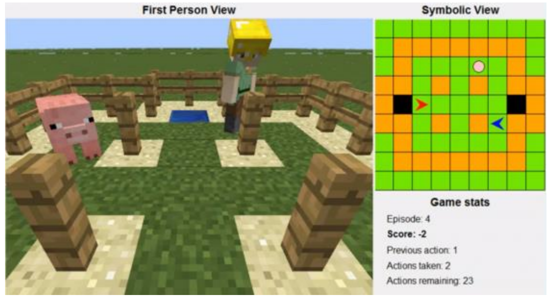
> Computationally more complex than Chess and Go
  > Multi-objective tasks to reach goal
  > Multiple dissimilar units
  > Partial information
  > Long time horizons
  > Continuous action space and environment
> Require teamplay, depending on the type of game
  > Learning cooperation and teamplay is a paradigm shift for agents
  > Opens a host of real-world applications

Recent AI Advances in Video Games

- 2013 Atari DQN
- DeepMind
- Non-cooperative



- 2018 Quake III Arena CTF
- DeepMind
- Hybrid



- 2015 Malmo Minecraft
- Microsoft
- Cooperative

OpenAI Five

- 2018 Five Dota
- OpenAI
- Hybrid

Enter StarCraft II

Screenshot from DeepMind’s pysc2 API

Python for StarCraft II

- 2017: release of the pysc2 and python-sc2 packages for StarCraft II
- pysc2 (DeepMind)
  - Vinyals, O., et al. 2017. StarCraft II: A New Challenge For Reinforcement Learning
  - Geared towards building advanced reinforcement learning agents
- python-sc2 (Dentosal)
  - Geared towards ease of use for building both rule-based and AI agents

A Basic StarCraft II Agent

> 'Worker rush'
> Simple example of a rule-based agent < 20 lines of code
> Take everything you have at time t_0 and attack the enemy

> At t_0:
  > Select all workers w_0
  > Send w_0 to attack enemy start location

Code for Worker Rush

> Taken from https://github.com/Dentosal/python-sc2#example

```
> import sc2
> from sc2 import run_game, maps, Race, Difficulty
> from sc2.player import Bot, Computer


> class WorkerRushBot(sc2.BotAI):
      async def on_step(self, iteration):
          if iteration == 0:
              for worker in self.workers:
                  await self.do(worker.attack(self.enemy_start_locations[0]))

> def main():
      run_game(maps.get("Simple128"), [
         Bot(Race.Terran, WorkerRushBot()),
         Computer(Race.Protoss, Difficulty.Easy)
         ], realtime = True)

> if __name__ == '__main__':
      main()
```

StarCraft II – Agent

Properties of the SCII agent	
Behavior	Win game
Lifetime	Until defeat
Personality	Asymmetric, 3 different races, multiple 100s of units
Memory	Depends
Strategies	Balance resource management, expanding vs. defense

StarCraft II – Environment

Properties of the SCII environment	
Type	Real-time strategy (RTS) game
Gameplay	Fast paced micro-actions and need for high-level planning
Action space*	10^8, need for hierarchical actions
Environment states**	$10^{1,685}$
Rule of transition between states	Continuous, based on the agents' actions
Rewards assigned to each state	Unknown
Reward horizon	Long pay-off = strats more important than micro
Mode of information	Fog of war = imperfect information
Mode of action	Simultaneous

* Vinyals, O., et al. 2017. *StarCraft II: A New Challenge for Reinforcement Learning.* https://arxiv.org/abs/1708.04782
** Estimated for StarCraft Brood Wars. Usunier, N., et al. 2016. *Episodic Exploration for Deep Deterministic Policies: An Application to StarCraft Micromanagement Tasks.* https://arxiv.org/abs/1609.02993

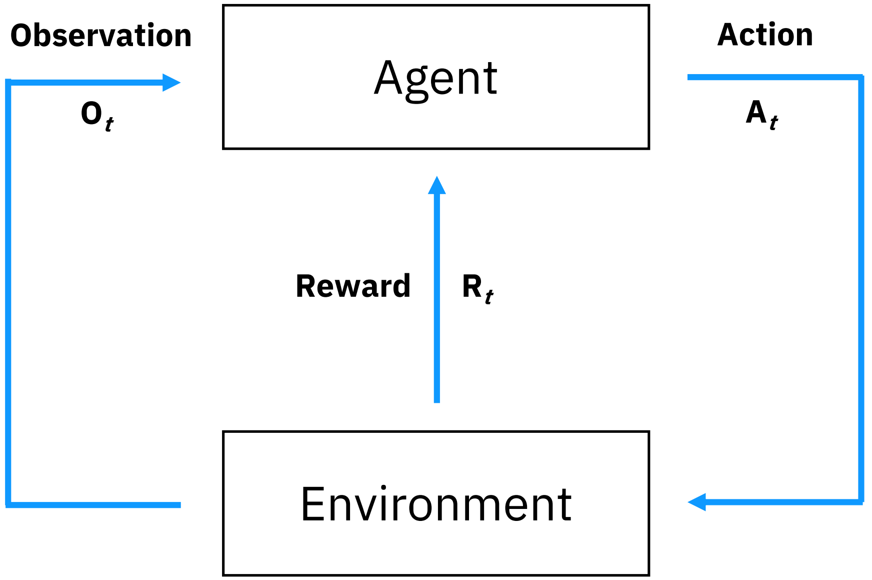
Agency in StarCraft

- M. Kim, S. Kim, K. Kim and A. Dey. 2016. Evaluation of StarCraft Artificial Intelligence Competiotion Bots by Experienced Human Players

- Five categories for evaluation
  - Production – produce units and buildings massively and effectively
  - Micro Management – control individual units
  - Combat – control armies to win combat
  - Decision Making – strategy and tactics under uncertainty
  - Performace – overall gameplay

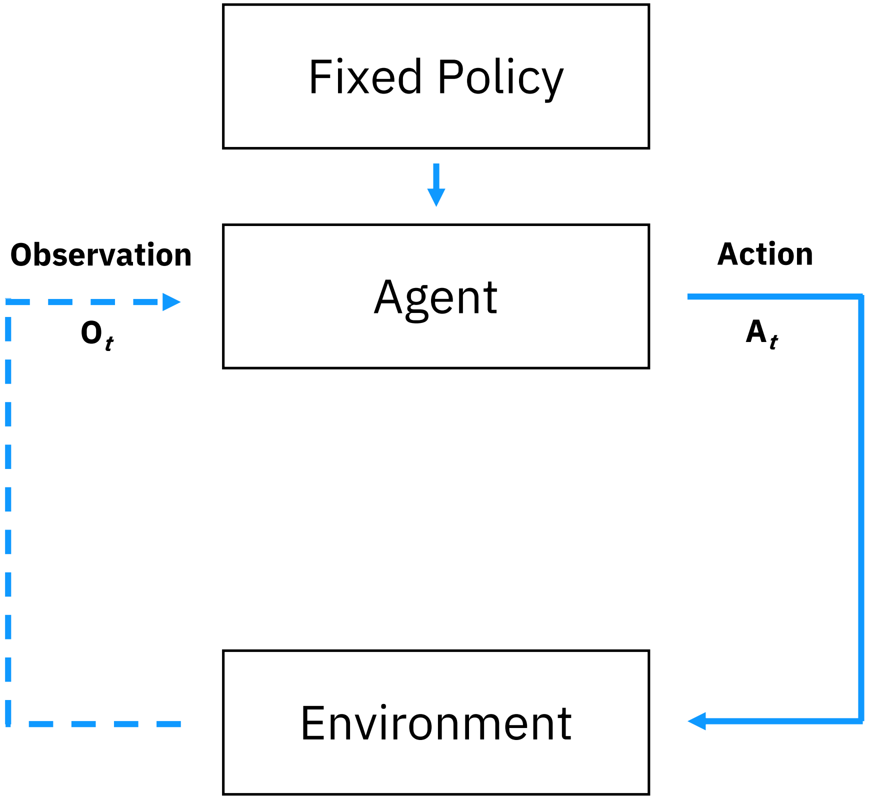
- Humans give greatest weight to micro management and decision making
- AI competitions won by bots with high combat and performance scores

Agent-Environment Loop 1



- At each step t the agent
  a. Exectues action A_t
  b. Receives observation O_t
  c. Receives scalar rewar R_t
- The environment
  a. Reveives action A_t
  b. Emits observation O_{t+1}
  c. Emits scalar reward R_{t+1}
- *t* increments at environment step

Agent-Environment Loop 2



- > At each step t the agent
  a. Exectues action A_t based on fixed policy
  b. Receives observation O_t
- > The environment
  a. Reveives action A_t
  b. Emits observation O_{t+1}
- > *t* increments at environment step
- > Observations restricted by fixed policy

Differences between Loop 1 and Loop 2

> Loop 1

a. Reward enables utility max.
b. Utility max. enables learning
c. Learning enables dynamic actions
d. Dynamic actions enable adaption
e. Reinforcement learning

> Loop 2

a. No reward, hence no utility
b. No utility, hence no learning
c. No learning, hence static actions
d. Static actions, hence no adaption
e. Rule-based

Respective Pros and Cons

- Loop 1
  a. Accommodates complex actions
  b. Requires meaningful rewards
  c. Underperforms in ‘mute’ env.
  d. Weakly predictable actions

- Loop 2
  a. Complexity constrained by coded rules
  b. Does not depend on rewards
  c. Constant actions across environments
  d. 100% predictable actions

- In theory, learning > rule-based. In practice, heavily dependent on environment.

Hard Trade-Offs

- Assuming that you have only limited knowledge about the environment and only one shot at success, which approach would you chose? Why?
- Would your answer change if you had more than one shot at success?

Framing the Trade-Offs

- The global implementation of AI brings about a host of questions
- AI Policy and AI Strategy have emerged as fields to provide answers
- **AI Policy**
  - Analysis and practive of societal decision-making about AI
- **AI Strategy**
  - Long-term conceptual analysis of AI developements
- While AI researchers recoginze IR as a valuable discipline, the IR community has yet to leverage its potential.

Challenge

> We need to integrate both the computational tools and the debate on implementing AI into IR thinking
> What are useful points of contact between AI and IR?
> Where do you see overlap and potential for applications?

Summary

> AI is having an increasing impact on the global order

> Stakeholders are looking for meaningful ways to respond

> Clear definitions of AI are needed to mitigate uncertainty

> The rational actor approach to AI offers high explanatory power

> Games and game theory can be used to develop AI agents

> Current video games present highly complex challenges for AI

> One of the games with the most strategic depth is StarCraft II

> Two Python APIs allow building AI and rule-based agents for StarCraft II

> The IR community needs to integrate these tools into its thinking and practice

Required Software

- For next week you only need Python. We will install more stuff along the way.
- Python **3.6 < version < 3.7** to ensure all packages work
  - https://www.python.org/downloads/
- IDLE (Python's default IDE) is sufficient
- For a more advanced IDE, download the community version of PyCharm
  - https://www.jetbrains.com/pycharm/download
  - Tutorial on setting up PyCharm: https://www.youtube.com/watch?v=0y5XlNeFxNk