Learning from Disagreement

Albert Tseng - SURF 2020

Introduction - Domain knowledge in Machine Learning

- Inefficient to learn without domain knowledge
  - e.g. self driving cars, labeled datasets
- Problem: domain experts are expensive and domain experts are humans
- Expensive
  - What is the right amount of domain knowledge needed? And in what format?
  - How to best use given domain knowledge
- Humans
  - What happens when domain experts disagree?
  - Is there information to be gained from disagreement?

Introduction - Domain knowledge in Machine Learning

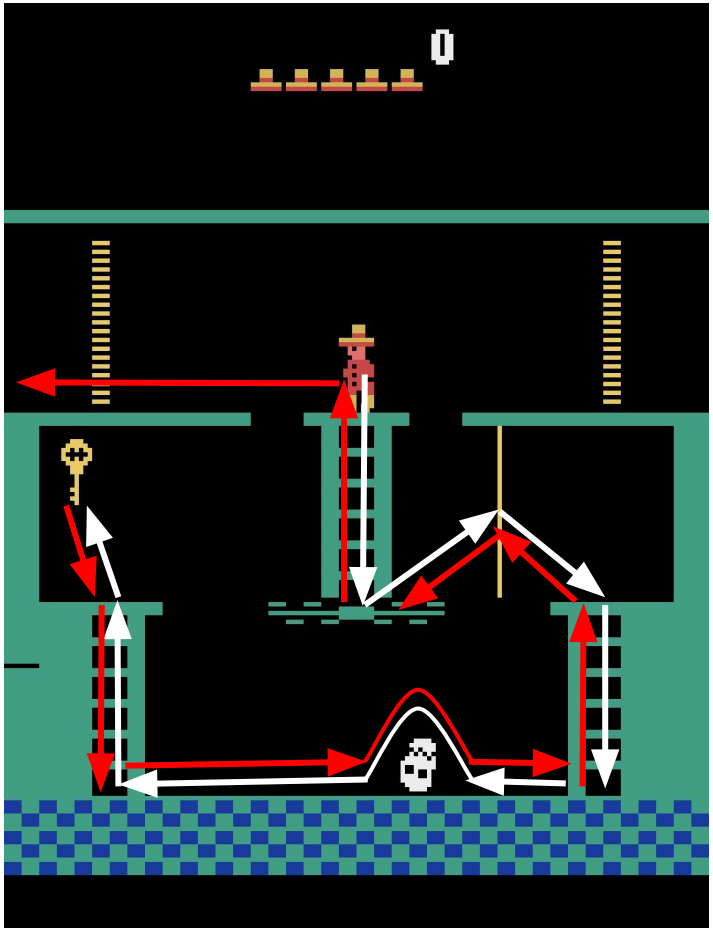
- Inefficient to learn without domain knowledge
  - e.g. self driving cars, labeled datasets
- Problem: domain experts are expensive and domain experts are humans
- Expensive
  - What is the right amount of domain knowledge needed? And in what format?
  - **How to best use given domain knowledge**
- Humans
  - What happens when domain experts disagree?
  - **Is there information to be gained from disagreement?**
- **Goal: utilize human disagreement to improve learning performance.**

Background - Problem Setting

- Reinforcement learning
  - Learning setting where an agent learns and performs a policy in an environment.
  - In goal conditioned RL, the agent is given a set of goals within the environment to satisfy.
- Goal conditioned learning
  - Class of algorithms that learn from goals as well as states.
  - Goals are a useful way of balancing tractability of multiple human labelers while directly impacting the performance of the model.
  - Existing approaches (Data Programming, Ratner et al.) are limited to statistical analyses due to computational cost with large datasets.
- Domain Knowledge and Goals
  - Labelers for goals
  - Each labeler gives signal for goal (reached or not) given state of agent
  - Each labeler has different definition for each goal, creating disagreement among labelers for a given goal.

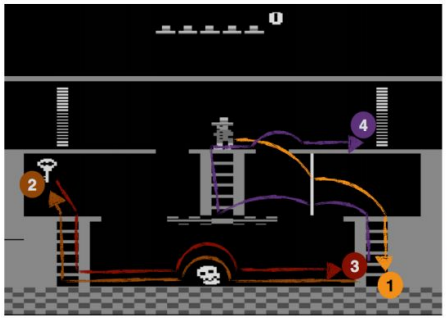
Background - Environment

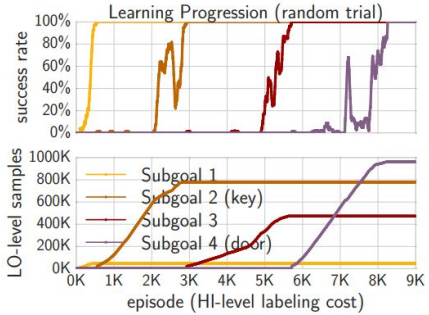
- Montezuma's Revenge (Atari 2600) Level 1
- Very difficult to solve with vanilla RL/IL methods (H-DQN, Kulkarni et al.)
- Only 2 sparse rewards
  - Get key (+100)
  - Exit through either door with key (+300)
- Easy for agent to die
  - Die if fall onto checkerboard region
  - Die if touches moving skull
- Optimal solution follows highlighted path to get key (first white to get key then red).



Prior Work - Hierarchical RL (H-DAGGER, Le et al.)

- Approach by to separate out RL agent from goal proposition to agent.
- Goals are proposed to agent for completion by an external metacontroller.
  - Agent gets small reward when it reaches goal
- Using a DQN agent and 4 well distributed subgoals, able to learn Montezuma's stage 1 with under 3M samples
- Not very stable, sensitive to goal initialization.





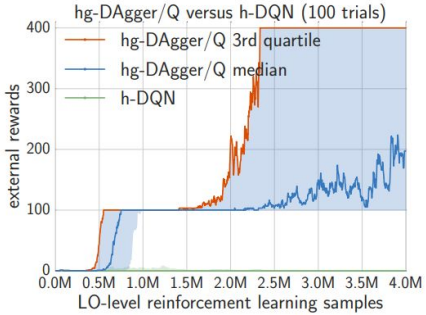


Image: Le et al.

Approach - Notation

- Given a set of goals $G = \{g_i\}$ and labelers $L = \{l_i\}$, we have a total set of label functions $Y = L \times G$.
  - Each label function can be applied to a trajectory t in dataset D to get a set of $|t|$ labels for t.
  - Each label corresponds to one timestep in the trajectory, with True if the agent satisfied the goal and False otherwise.
  - Y_{ijk} indicates a label from l_i for g_j at timestep k.
- We define the disagreement X_i for a goal g_i as follows:
  - Essentially, disagreement is 1 - the average of % labelers labeling True for each timestep given at least one labeler labels true for included timesteps
- True signal from any labeler = true for agent

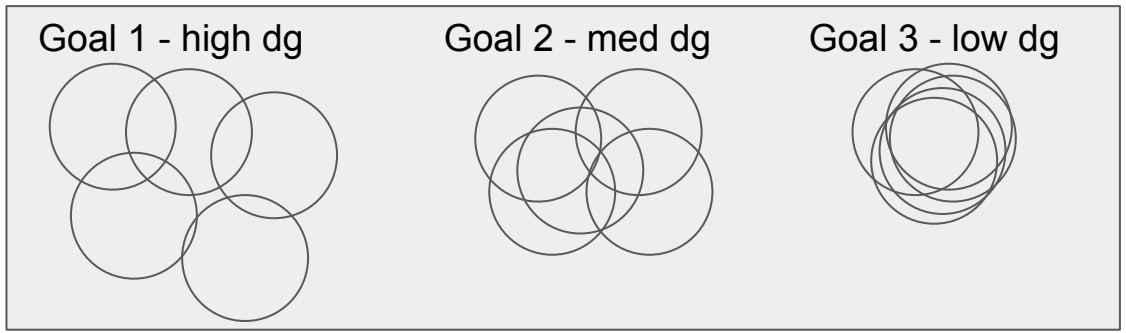
$$X_i = \frac{1}{N} \sum_{t \in D} \sum_{k=1}^{|t|} \frac{|L| - A_t}{|L|} * \mathbb{1}_{A>0}$$

$$A_t = \sum_{l=1}^{|L|} Y_{lit}$$

$$N = \sum_{t \in D} \sum_{k=1}^{|t|} 1$$

Approach - Curriculum Learning

- Intuition: goals with high disagreement are “easier” to learn
- High disagreement for g_i implies a large portion of timesteps do not have unison among labelers
- Translated to the state space, the set of states that result in true rating for at least one labeler is larger for higher disagreement
  - Assuming uniform prior over area of the “true” state space for all labeler/goal combos

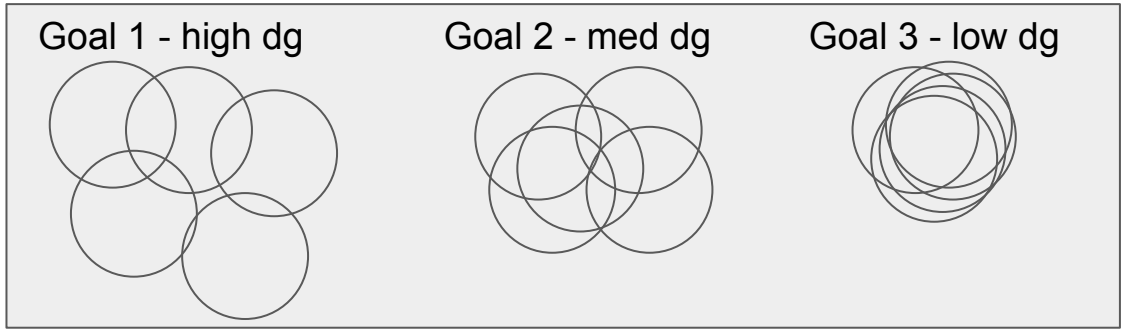


Each goal/labeler “true” state space is represented by a circle.

High disagreement goals (L) cover a larger union compared to low disagreement goals (R)

Approach - Curriculum Learning

- Train metacontroller to map state -> valid goals, conditioned on disagreement
- During training, schedule disagreement decay and propose goals accordingly
- Learn high disagreement (easier) goals first, then low disagreement (harder) goals.

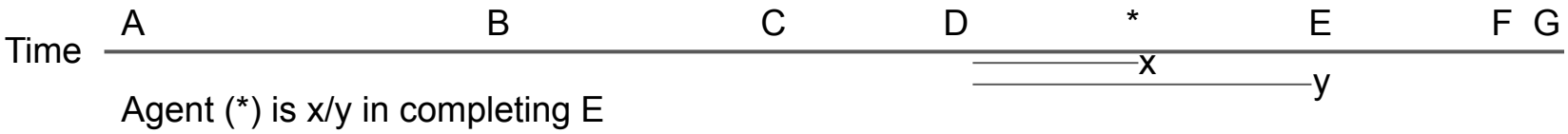


Each goal/labeler “true” state space is represented by a circle.

High disagreement goals (L) cover a larger union compared to low disagreement goals (R)

Approach - Dense Rewards

- Main issue with Montezuma's revenge is sparsity of rewards
  - Only two rewards given (get key +100, finish stage +300)
- Giving reward for goals helps, but limited by number of goals
  - If goals are not well distributed, rewards still sparse
- Learn model that predicts, given state, how close % wise agent is to target goal
  - % there measured by timesteps in expert demonstrations
  - Maximum % taken over labelers for goal
- Dense reward fed as Δ % there sampled every n timesteps



Experiments

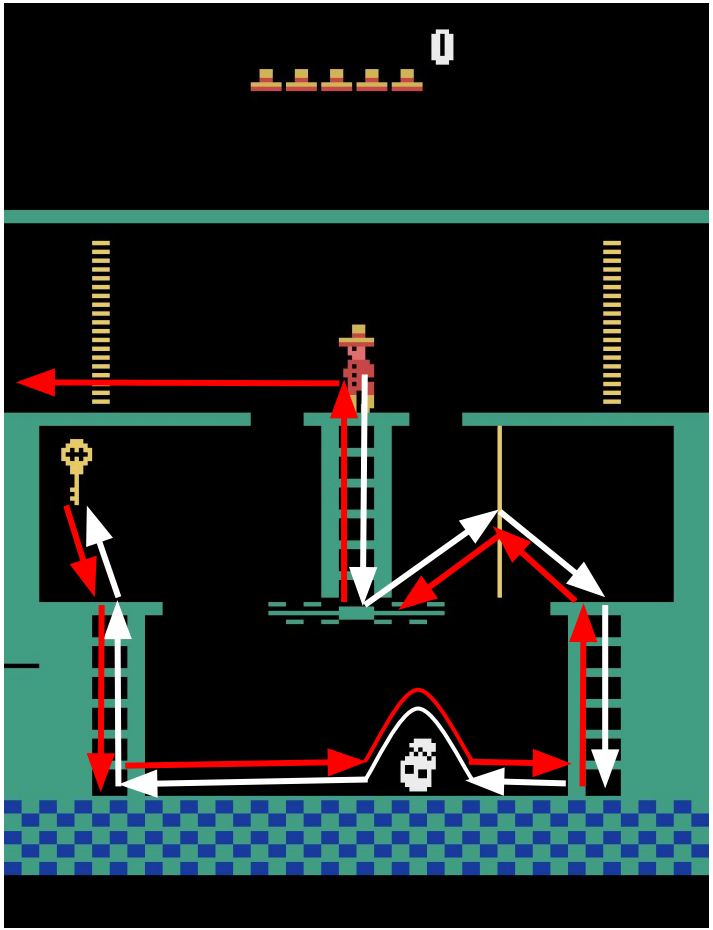
- 4 goals in Montezuma's Revenge, 3 labelers each

Goal	Disagreement	Definition
Avoid Dying	0.186	Get to top of RHS stairs without falling in checkerboard pit (agent dies)
Avoid Skull	0.425	Avoid hitting the moving skull on the floor
Get Key	0.154	Get the key
Finish	0.616	Finish the Stage

- Similar hierarchical metacontroller + low level agent as H-DAGGER
- Agent is DQN agent with experience replay
- Metacontroller is disagreement conditioned CNN
- Dense reward model (one per goal/labeler combination) is CNN
- Run experiments on baseline (H-DAGGER) and Dense Reward + Curriculum

Background - Environment

- Montezuma’s Revenge (Atari 2600) Level 1
- Very difficult to solve with vanilla RL/IL methods
- Only 2 sparse rewards
  - Get key (+100)
  - Exit through either door with key (+300)
- Easy for agent to die
  - Die if fall onto checkerboard region
  - Die if touches moving skull
- Optimal solution follows highlighted path to get key (first white to get key then red).

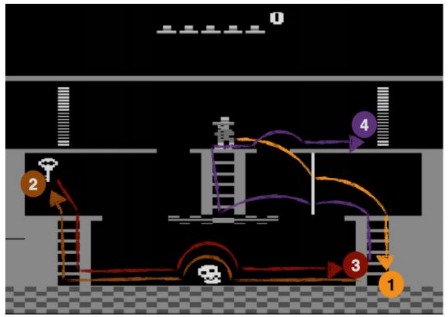


Results

- 4 goals in Montezuma's Revenge, 3 labelers each

Goal	Disagreement	Definition
Avoid Dying	0.186	Get to top of RHS stairs without falling in checkerboard pit (agent dies)
Avoid Skull	0.425	Avoid hitting the moving skull on the floor (towards key and away)
Get Key	0.154	Get the key
Finish	0.616	Finish the Stage

- H-DAGGER cannot learn past Avoid Dying[1]
  - High sparsity between "Avoid Dying" and "Avoid Skull"
  - Agent gets stuck in trying to reach "Avoid Skull"
- Dense Reward model able to reach 91% success rate on "Avoid Skull" and 75% success rate on "Get Key" with 1M samples[2]
  - Fails to pass "Avoid Skull" on return trip



1, 2) 3 trials for both; each trial takes multiple days so a large number of trials are not currently possible.

Conclusions and Future Work

- Curriculum + Dense Reward appears to work better than a baseline Hierarchical RL approach
- Still not able to complete Level 1 consistently; product of goal choice
- Test other approaches to tying disagreement to dense rewards
  - Beyond simple union of labelers
- Test curriculum on more complex environment
  - Minecraft MineRL environment; significantly more choices
- Generate intermediate goals from utilizing labeler disagreement and existing goals
  - Create denser distribution of goals and rewards

Acknowledgements

Thanks to Prof. Yisong Yue and Adith Swaminathan (MSR) for advising this project.

This project was made possible with support from Samuel P. and Frances Krown