Machine Learning

Machine Learning: Jordan Boyd-Graber
University of Maryland
REINFORCEMENT LEARNING: POLICY SEARCH

Policy Search

- Problem: often feature-based policies that work well aren't those that approximate V/Q best
- Solution: Find policies that maximize rewards rather than the value that predicts rewards
- Successful

Example: Imitation Learning

- Take examples of experts $\{(s_1, a_1) \dots \}$
- Learn a classifier mapping $s \longrightarrow a$
- Create loss as the negative reward

Example: Imitation Learning

- Take examples of experts $\{(s_1, a_1) \dots\}$
- Learn a classifier mapping $s \longrightarrow a$
- Create loss as the negative reward
- What if we diverge?

How do we find a good policy?

- Find optimal policies through dynamic programming $\pi_0 \equiv \pi*$
- Represent states s through a feature vector $\vec{f}(s)$

How do we find a good policy?

- Find optimal policies through dynamic programming $\pi_0 \equiv \pi*$
- Represent states s through a feature vector $\vec{f}(s)$
- Until convergence:
  - Generate examples of state action pairs: $(\pi_t(s), s)$
  - Create a classifier that maps states to actions (an apprentice policy) $h_t : f(s) \mapsto A$
  - Interpolate learned classifier $\pi_{t+1} = \lambda \pi_t + (1-\lambda) h_t$

How do we find a good policy?

- Find optimal policies through dynamic programming $\pi_0 \equiv \pi*$
- Represent states s through a feature vector $\vec{f}(s)$
- Until convergence:
  - Generate examples of state action pairs: $(\pi_t(s), s)$
  - Create a classifier that maps states to actions (an apprentice policy) $h_t : f(s) \mapsto A$ (Loss of classifier is the negative reward)
  - Interpolate learned classifier $\pi_{t+1} = \lambda \pi_t + (1 - \lambda) h_t$

How do we find a good policy?

- Find optimal policies through dynamic programming $\pi_0 \equiv \pi*$
- Represent states s through a feature vector $\vec{f}(s)$
- Until convergence:
  - Generate examples of state action pairs: $(\pi_t(s), s)$
  - Create a classifier that maps states to actions (an apprentice policy) $h_t : f(s) \mapsto A$ (Loss of classifier is the negative reward)
  - Interpolate learned classifier $\pi_{t+1} = \lambda\pi_t + (1-\lambda)h_t$
- DAGGER: Dataset aggregation [Ross, Gordon & Bagnell, 2010]
- searn: Searching to Learn [Daumé & Marcu, 2006]

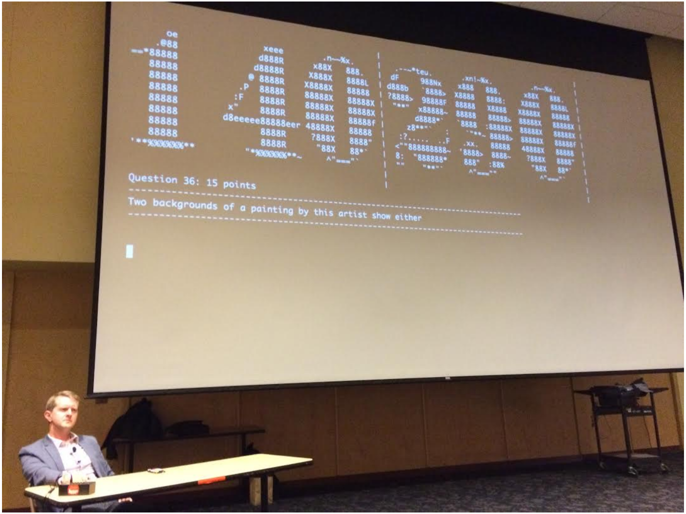
Applications of Imitation Learning

- Car driving
- Flying helicopters
- Question answering
- Machine translation

Applications of Imitation Learning

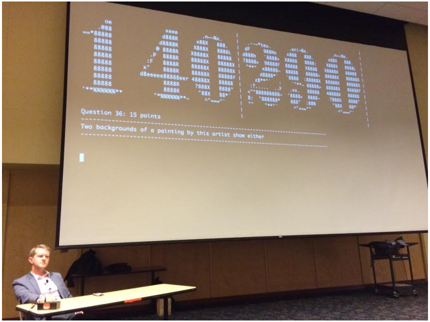
- Car driving
- Flying helicopters
- Question answering
- Machine translation

Question Answering



- **State:** The words seen, opponent

Question Answering



- **State**: The words seen, opponent
- **Action**: Buzz or wait
- **Reward**: Points

Why machine translation really hard is

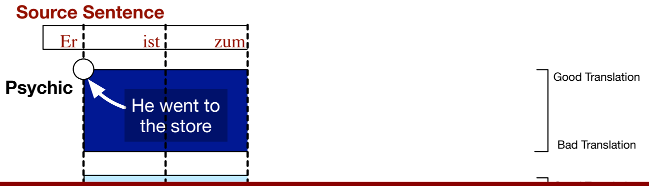
- **State**: The words you've seen, output of machine translation system
- **Action**: Take translation, predict the verb
- **Reward**: Translation quality

Comparing Policies

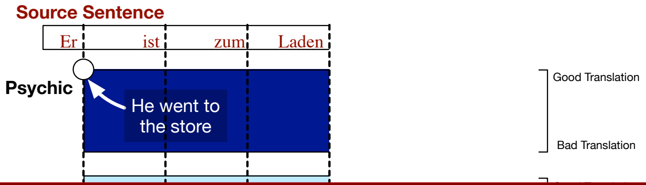
Comparing Policies

Comparing Policies

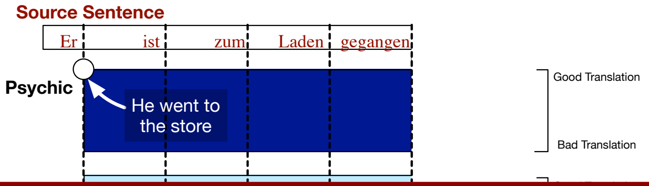
Comparing Policies



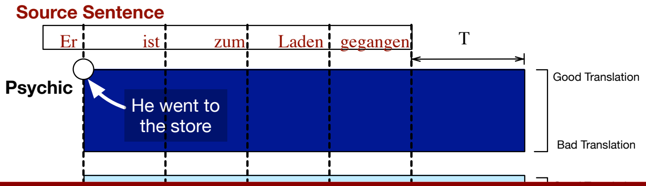
Comparing Policies



Comparing Policies

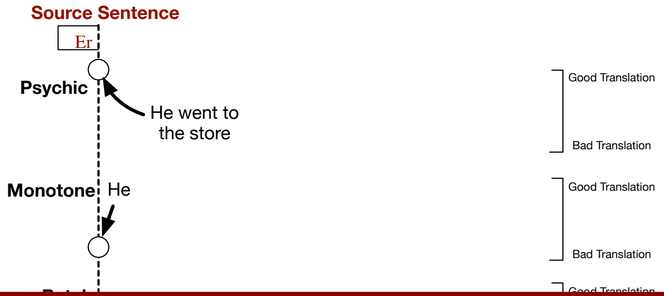


Comparing Policies

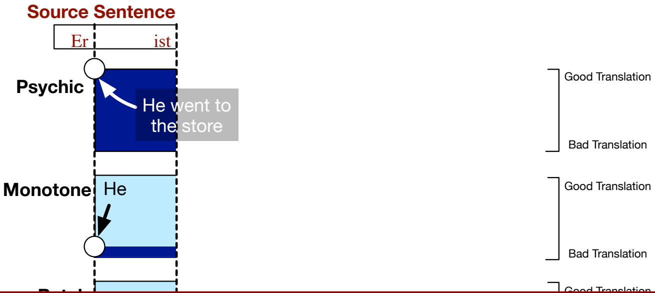


Comparing Policies

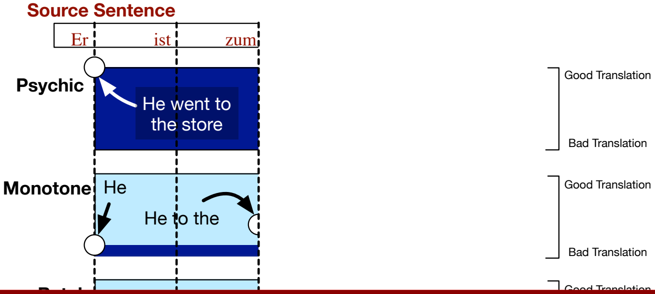
Comparing Policies



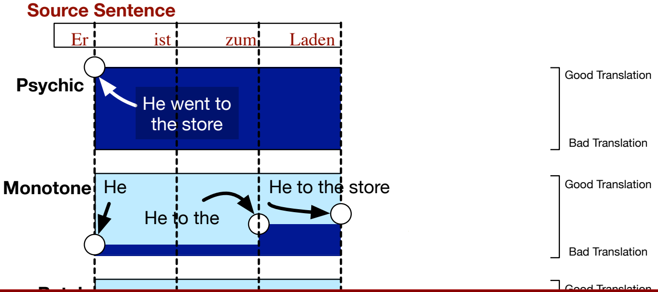
Comparing Policies



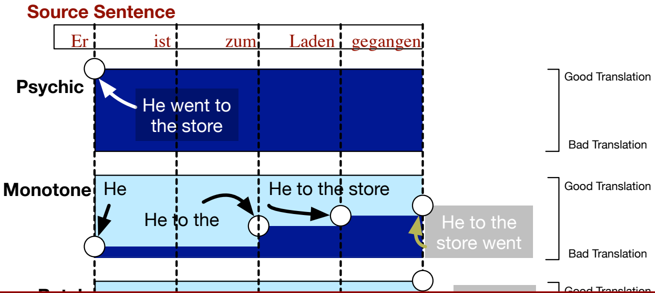
Comparing Policies



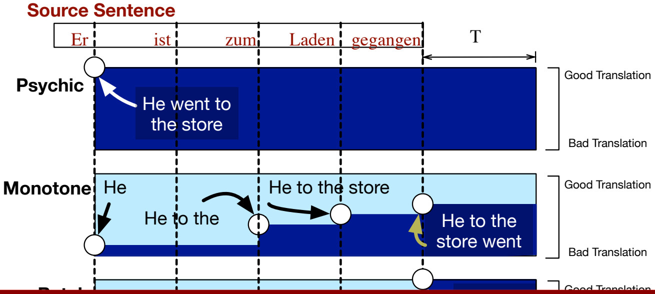
Comparing Policies



Comparing Policies

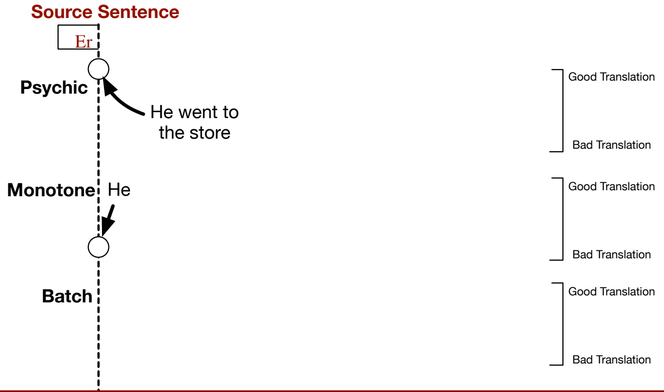


Comparing Policies

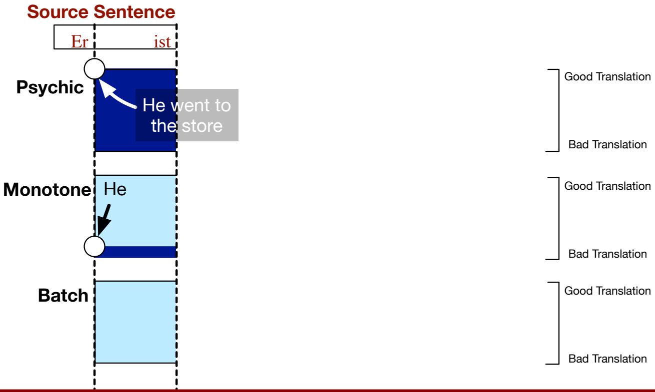


Comparing Policies

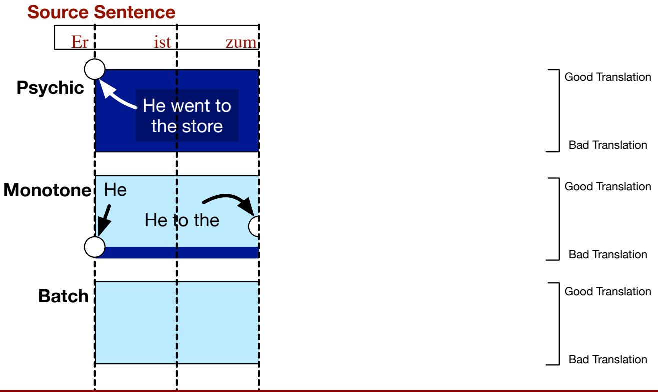
Comparing Policies



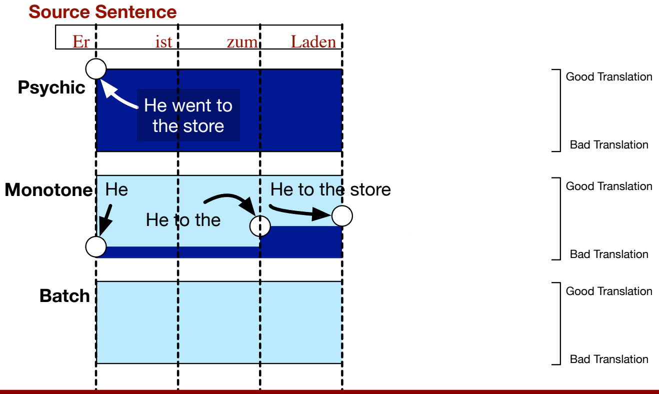
Comparing Policies



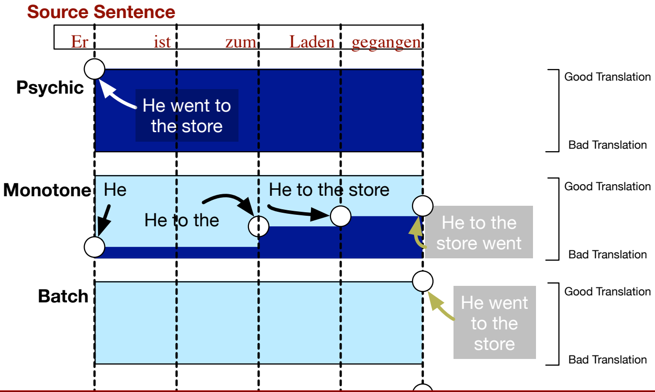
Comparing Policies



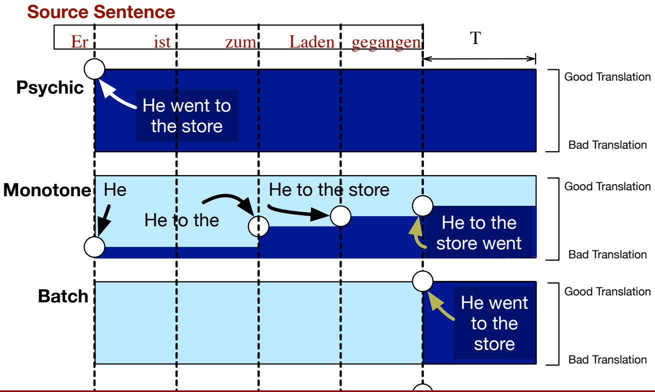
Comparing Policies



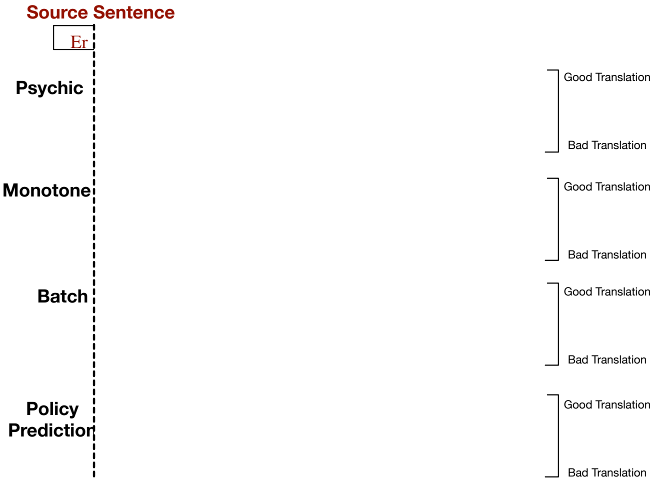
Comparing Policies



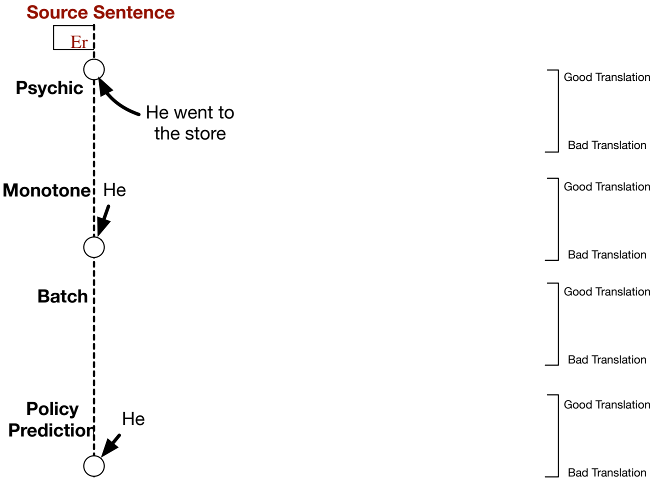
Comparing Policies



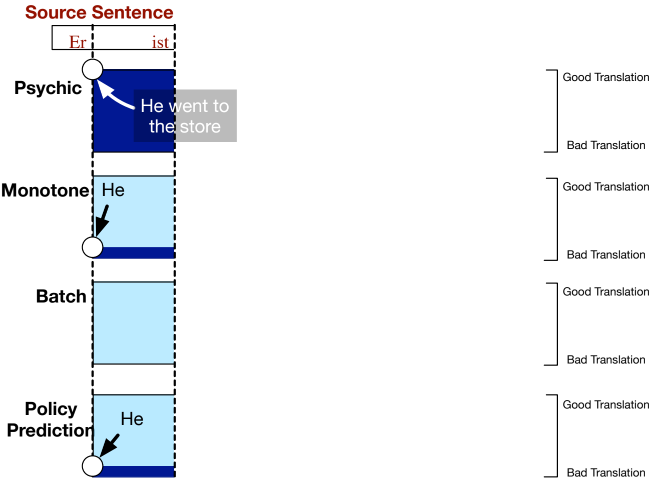
Comparing Policies



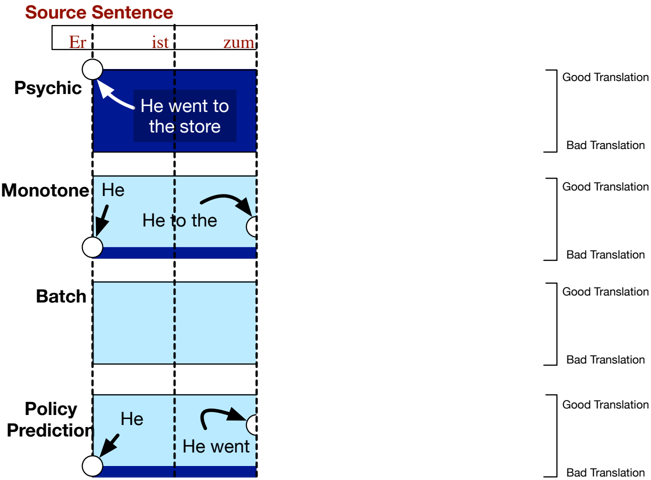
Comparing Policies



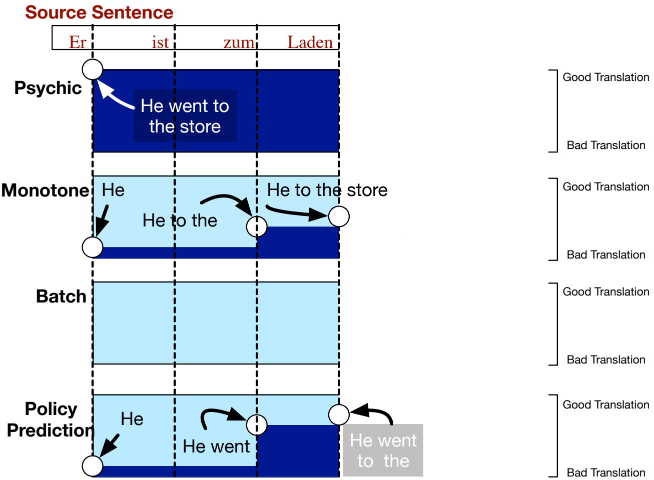
Comparing Policies



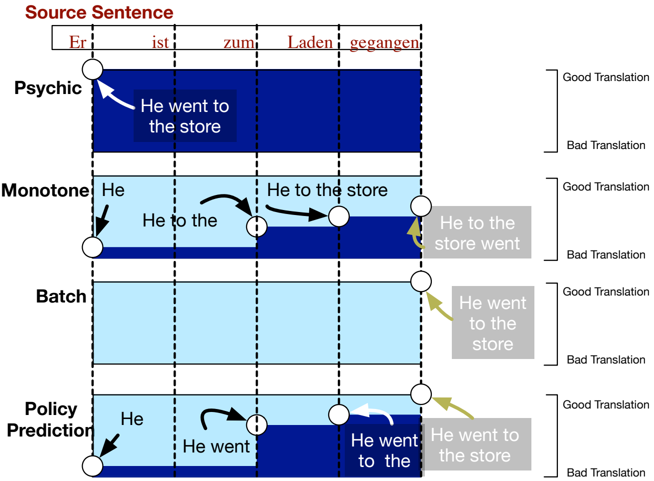
Comparing Policies



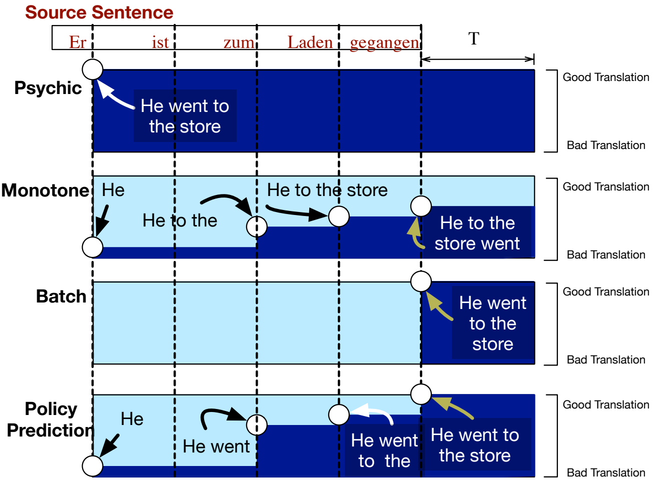
Comparing Policies



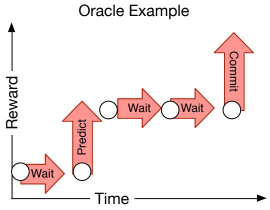
Comparing Policies



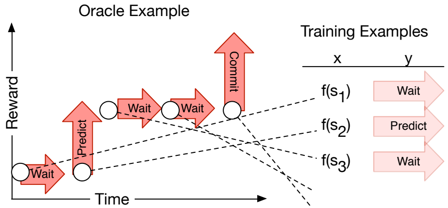
Comparing Policies



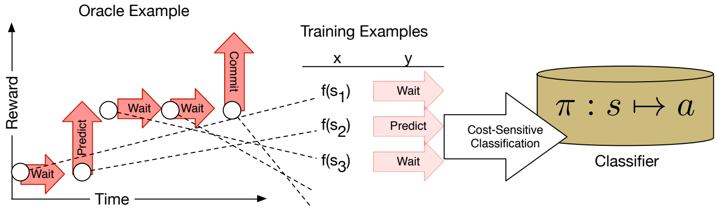
Applying SEARN



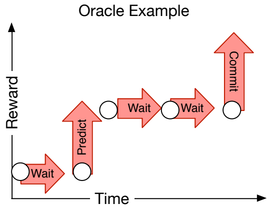
Applying SEARN

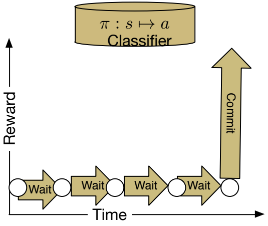


Applying SEARN

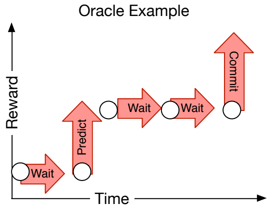


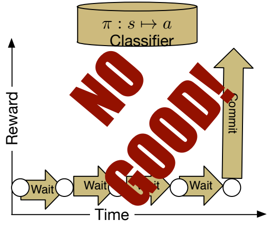
Applying SEARN



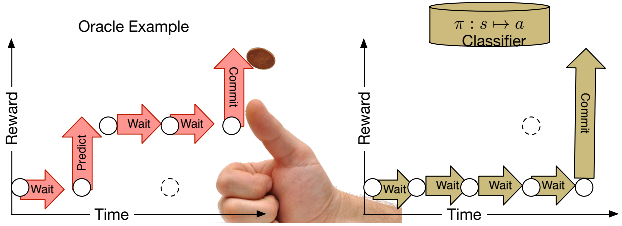


Applying SEARN

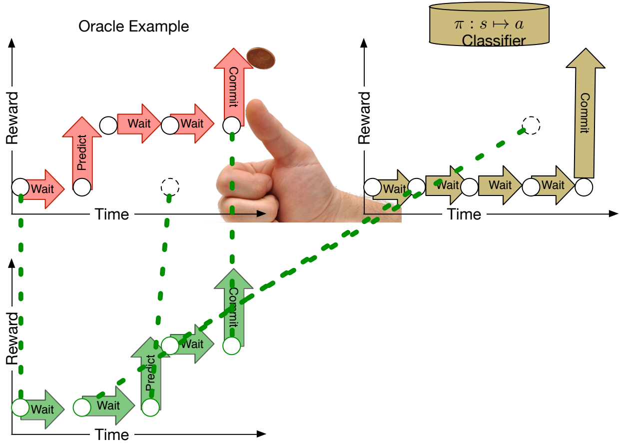




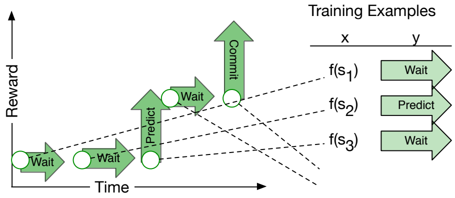
Applying SEARN



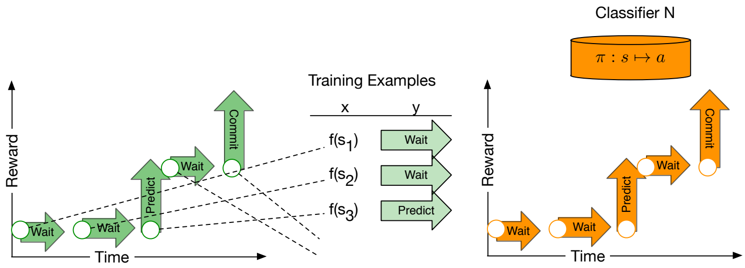
Applying SEARN



Applying SEARN



Applying SEARN



Recap

- Learning from examples: immitation learning
- Role of supervised machine learning
- Room for deep learning ...