

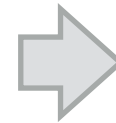
# Language grounding for cross-domain policy transfer and spatial reasoning

*Karthik Narasimhan*  
*OpenAI*

# Grounding semantics in control applications

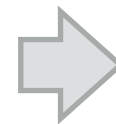
*Input*

***Text:***  
instructions or  
knowledge



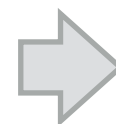
1. **Ghosts** chase and try to kill you
2. Collect all the **pellets**
3. ...

***Environment:***  
execute actions and  
observe



*Output*

***Control Policy:***  
sequence of actions  
to interact optimally



**MOVE LEFT  
STAY  
MOVE UP**

...

# Grounding semantics in control applications

I. Use language to improve performance in control applications



*Score: 7*



*Score: 107*

+

1. **Ghosts** chase and try to kill you
2. Collect all the **pellets**
3. ...

# Grounding semantics in control applications

## 1. Use language to improve performance in control applications



*Score: 7*



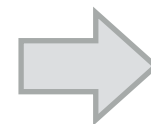
*Score: 107*

+

1. **Ghosts** chase and try to kill you
2. Collect all the **pellets**
3. ...

## 2. Use feedback from control application to understand language

Walk across the bridge



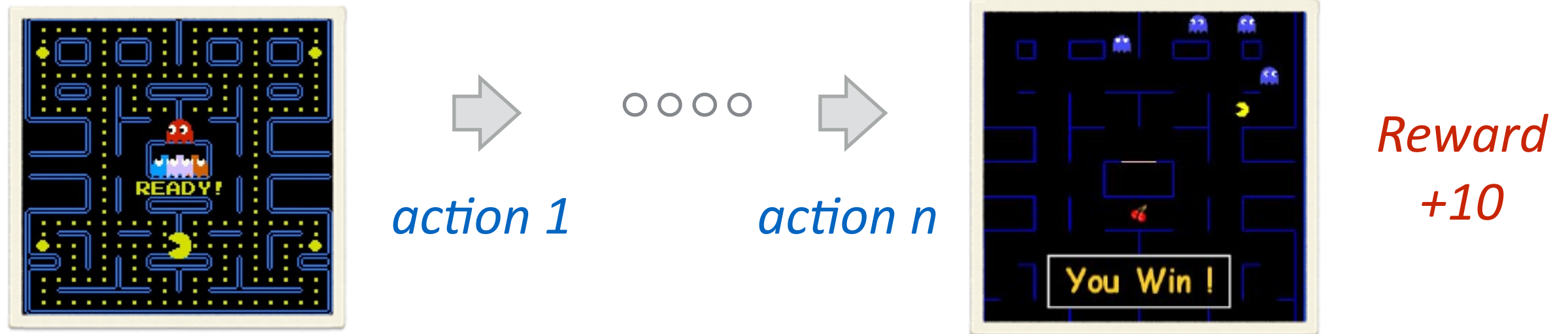
*Reward  
+1*

*Alleviate dependence on large scale annotation*



# Reinforcement Learning

- Delayed feedback



⇒ *How to perform credit assignment for individual actions*

- Large number of possible action sequences

⇒ *Need for effective exploration*

Improved language understanding translates  
to improved task performance

# Deep Transfer in Reinforcement Learning by Language Grounding

*Karthik Narasimhan, Regina Barzilay, Tommi Jaakkola*  
*MIT*

# Deep reinforcement learning for games



*Standard approach:* deep Q-learning by acting in the environment

*Steps to convergence:* ~ a few million

# Traditional RL framework

## Markov decision process

*State*  $s$  = Observed Environment

*Action*  $a$  = Move/Shoot/Use sword

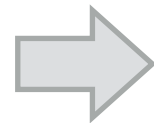
*State 1*

*Action*

*State 2*



MOVE RIGHT



*Reward*  
 $+1$

*Policy*

$$\pi : s \rightarrow a$$

*Action value function*

$$Q(s, a)$$



# Estimating a policy

Learn from sampled experiences

*Episode 1*



*Episode 2*

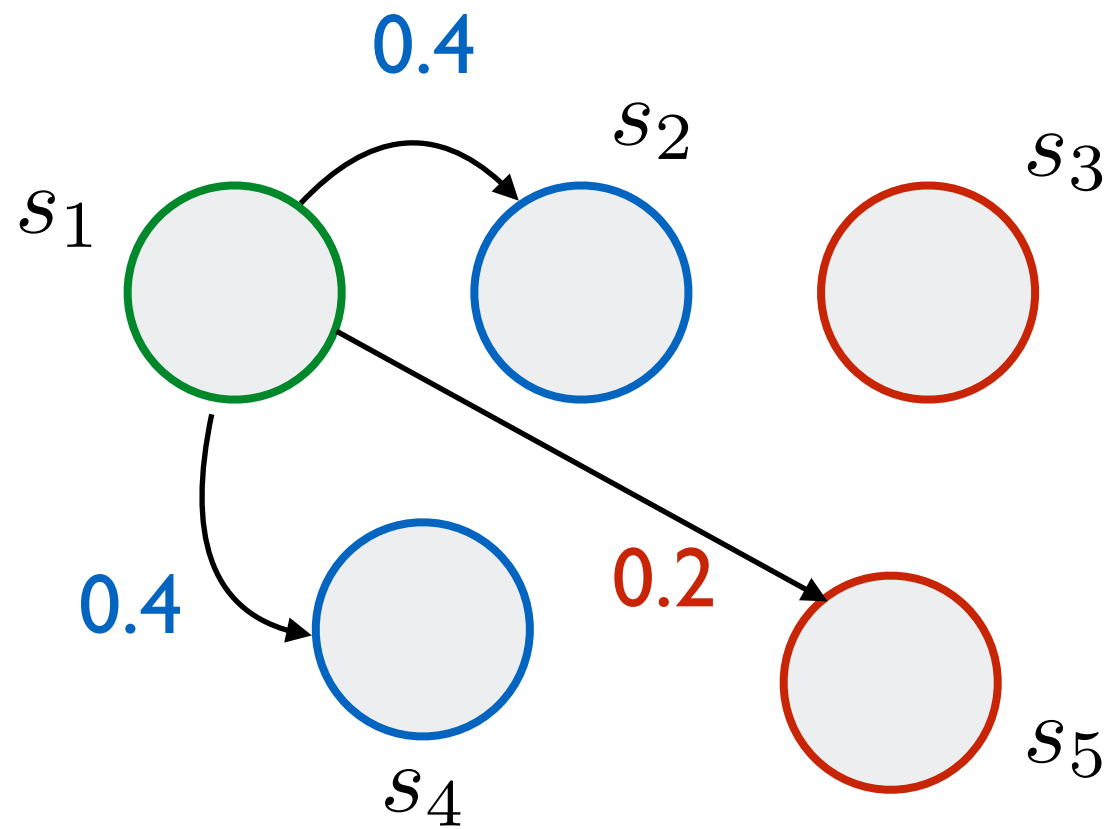


*Episode 3*



$$\pi(a|s)$$

# Environment state space



- Learn transitions between states
- Identify good vs bad states

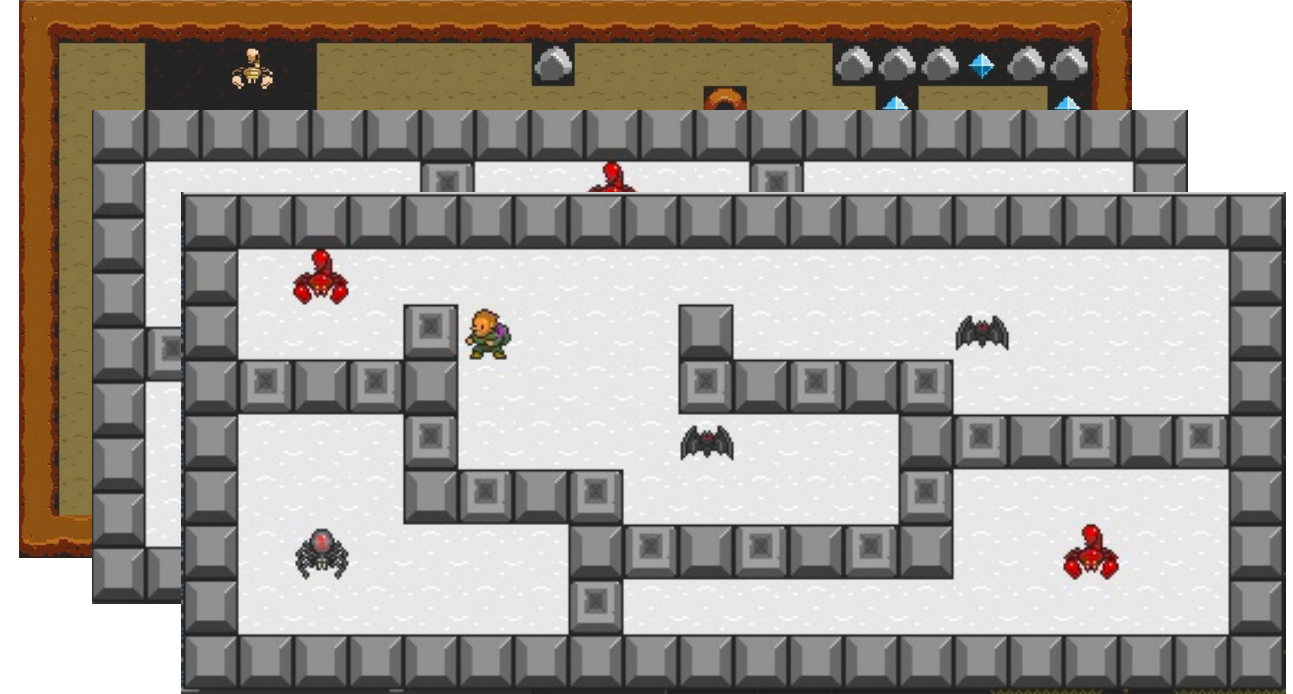


# More games



- Each new game requires re-learning from scratch

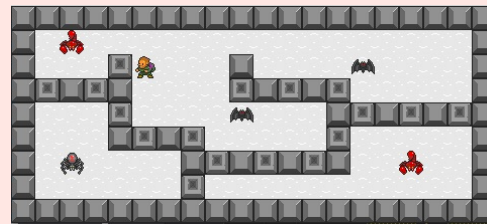
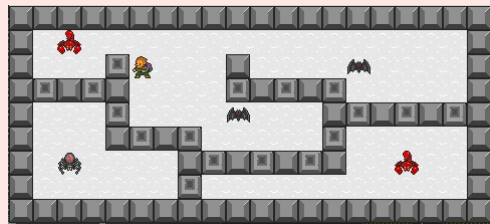
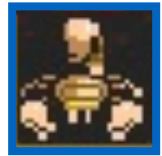
# More games



- Each new game requires re-learning from scratch
- Policy transfer: challenging

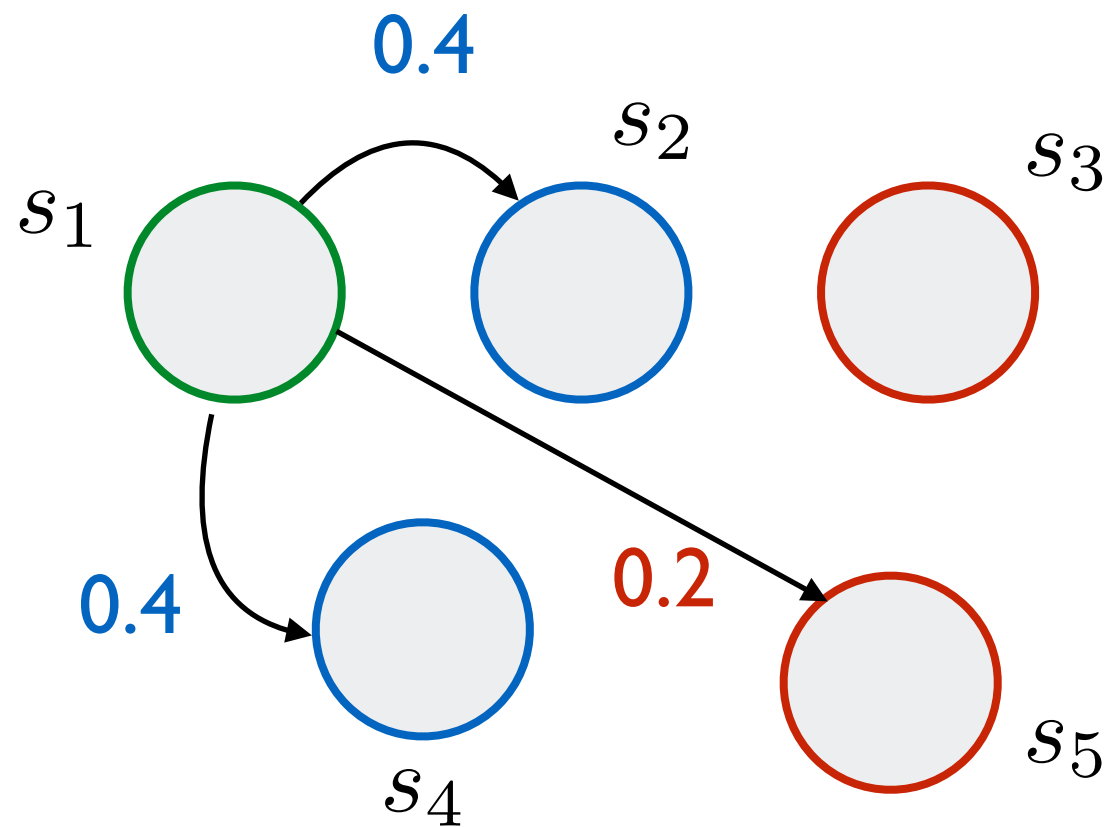
(Taylor and Stone, 2009; Parisotto et al., 2015, Rusu et al., 2016; Rajendran et al., 2017, ...)

# Why is transfer hard?

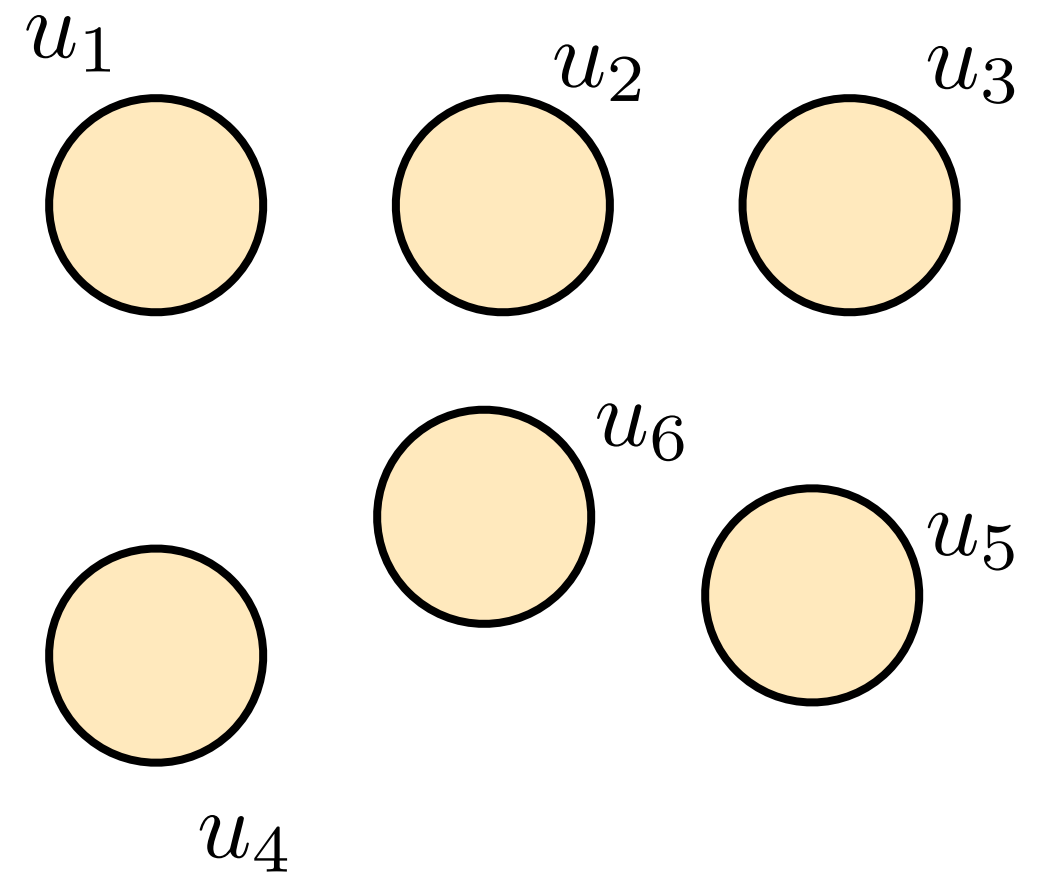


- Different state spaces and actions
- Need to explore new environment to learn mapping
- Incorrect mapping leads to negative transfer

## Environment 1

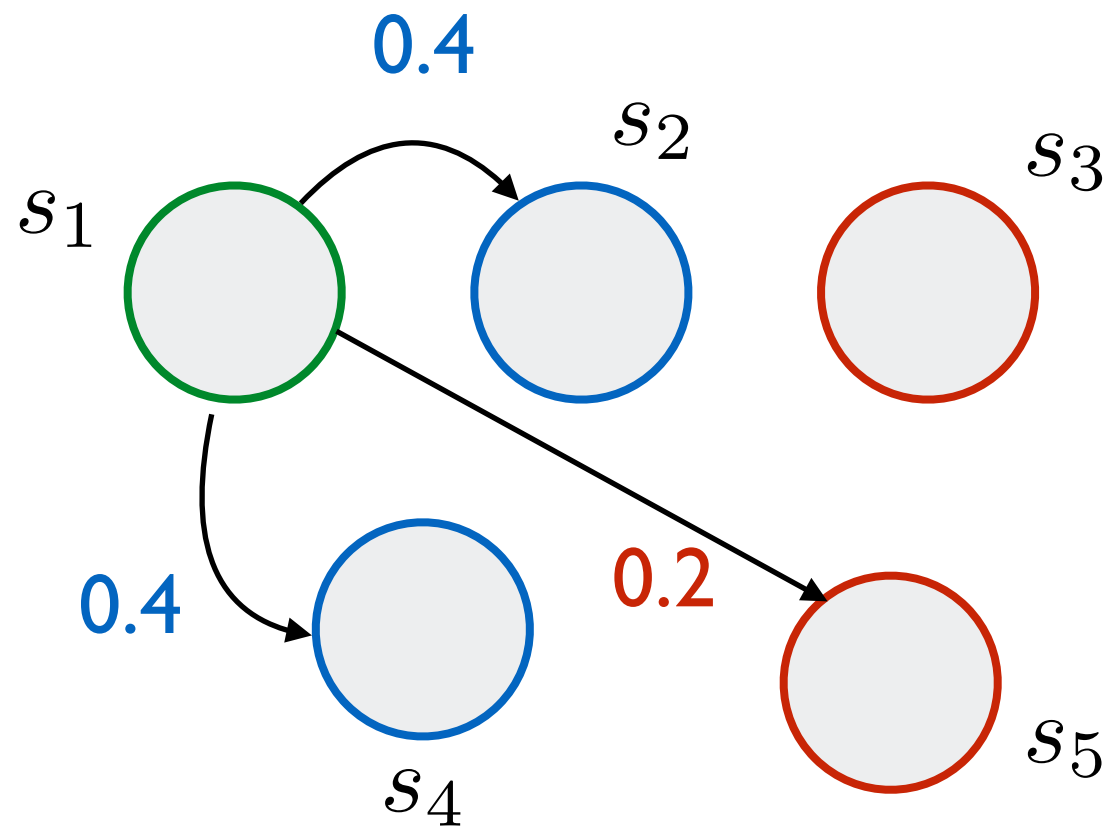


## Environment 2

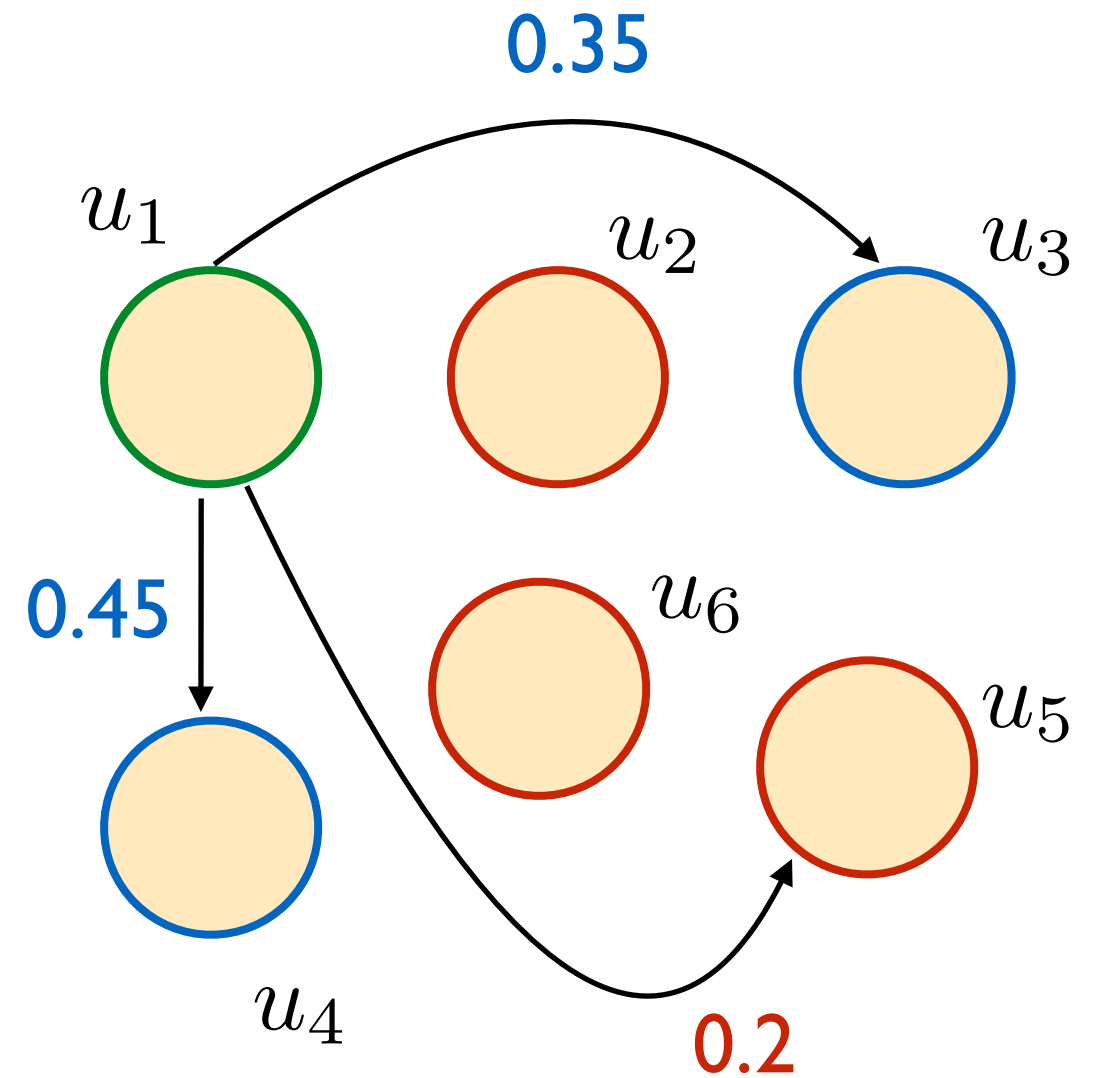


- How do we re-use learnt information?
- Need some anchor

Environment 1



Environment 2



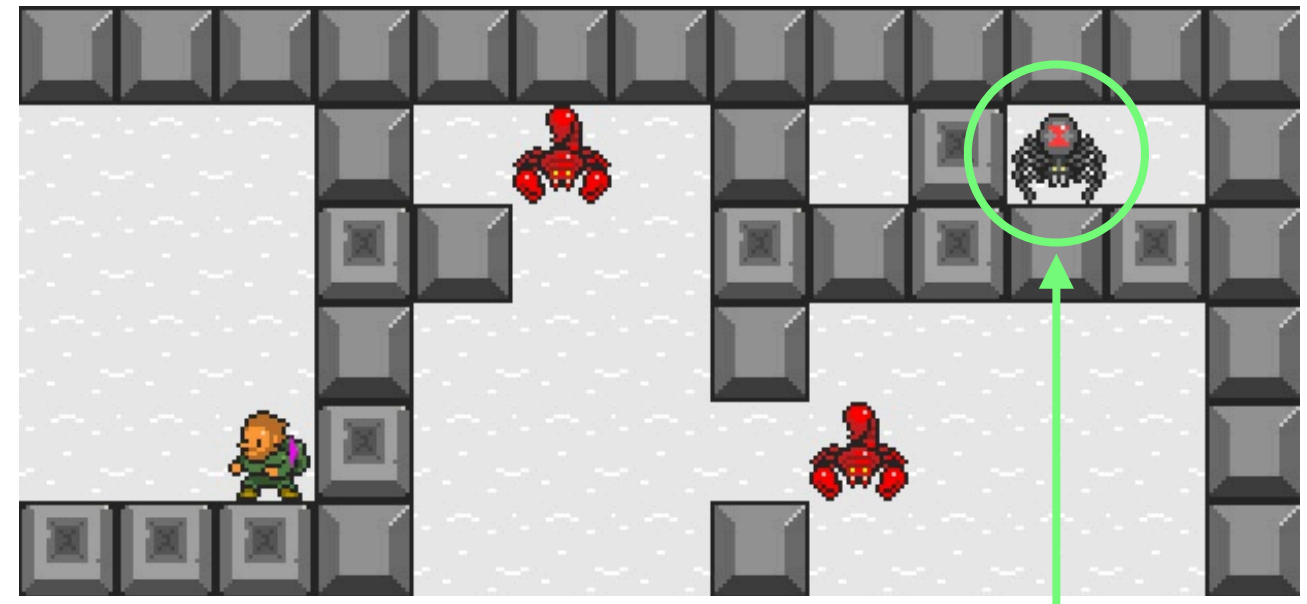
*$s_1$  is similar to  $u_1$   
 $s_2$  is similar to  $u_3$   
...*



# Using text descriptions



Scorpions chase you  
and kill you on touch

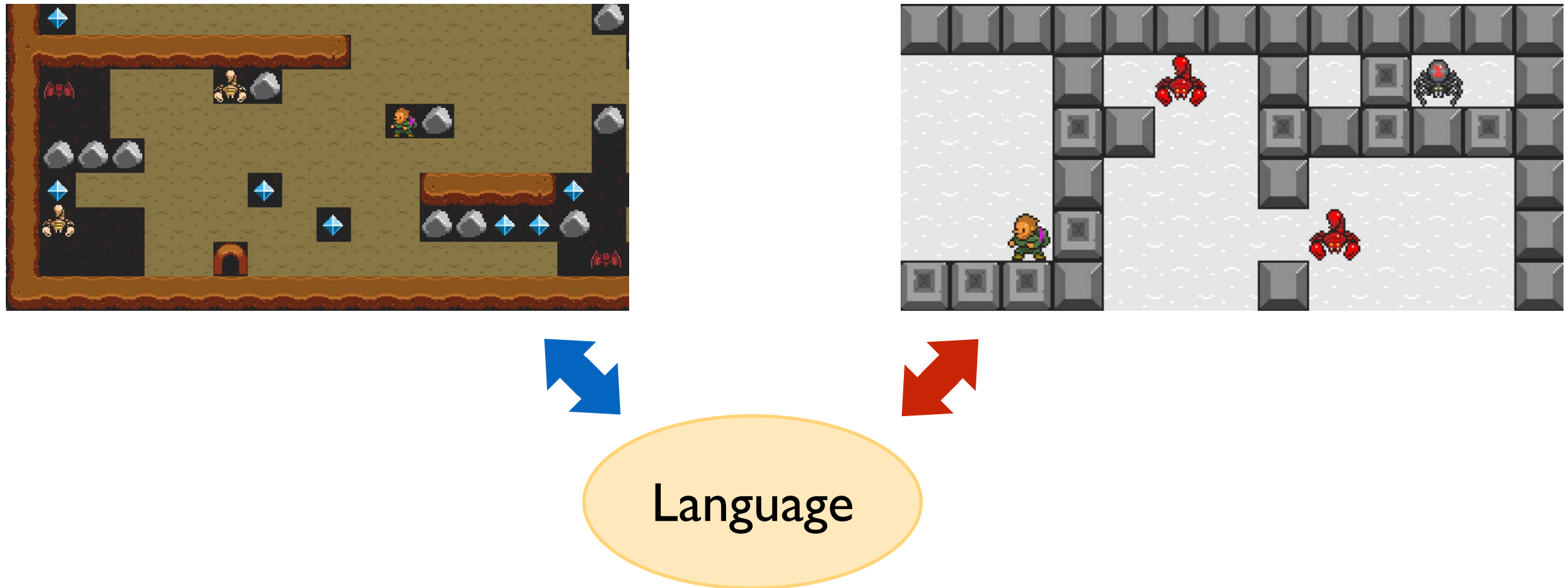


Spiders are chasers  
and can be destroyed  
by an explosion

- Text descriptions associated with objects/entities
- No mapping between objects in different environments



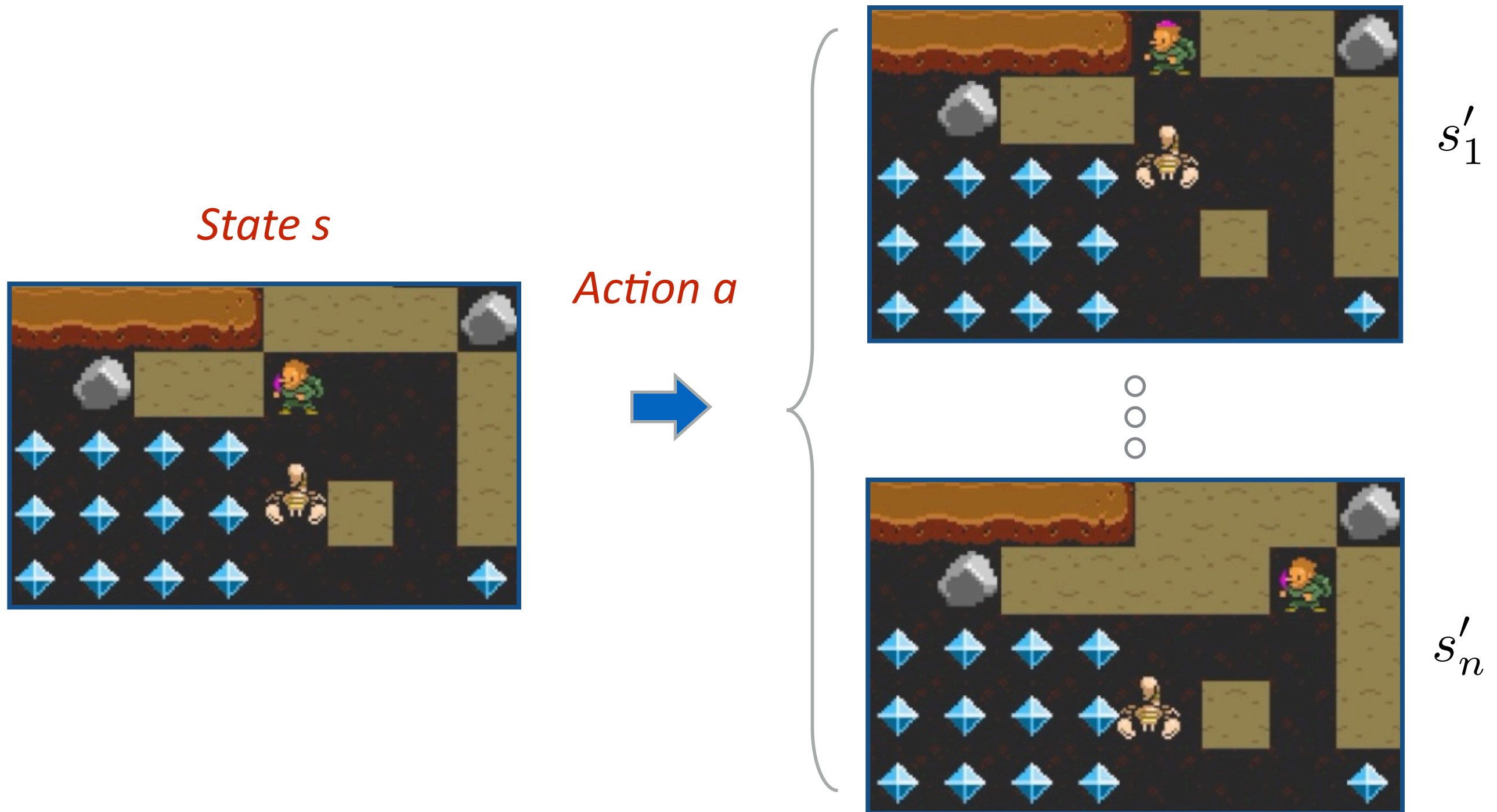
# Transfer through language



- Language as domain-invariant and accessible medium
- *Traditional approaches*: direct policy transfer
- *This work*: transfer ‘model’ of the environment using text descriptions

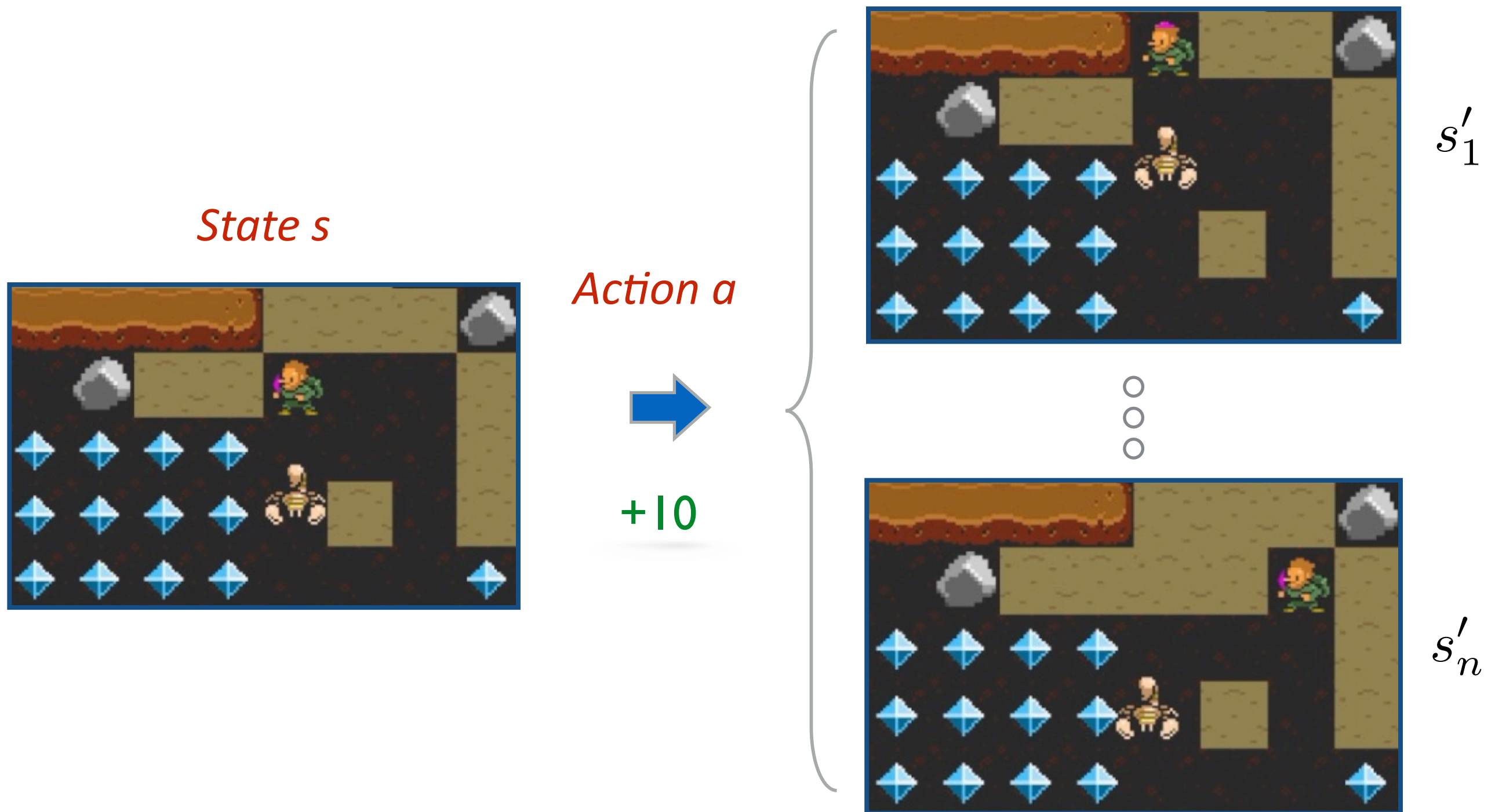
# Model-based reinforcement learning

Transition distribution  $T(s'|s, a)$



# Model-based reinforcement learning

Transition distribution  $T(s'|s, a)$  and reward function  $R(s, a)$



# Value Iteration

Action value function

$$Q(s, a) = R(s, a) + \gamma \sum_{s'} T(s'|s, a) V(s')$$

Value function

$$V(s) = \max_a Q(s, a)$$

Accurately estimating T and R is challenging

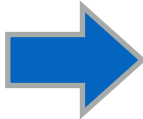


Text-conditioned transition distribution  $T(s' | s, a, z)$

State  $s$



Action  $a$



Scorpions chase you  
and kill you on touch

Text  $z$

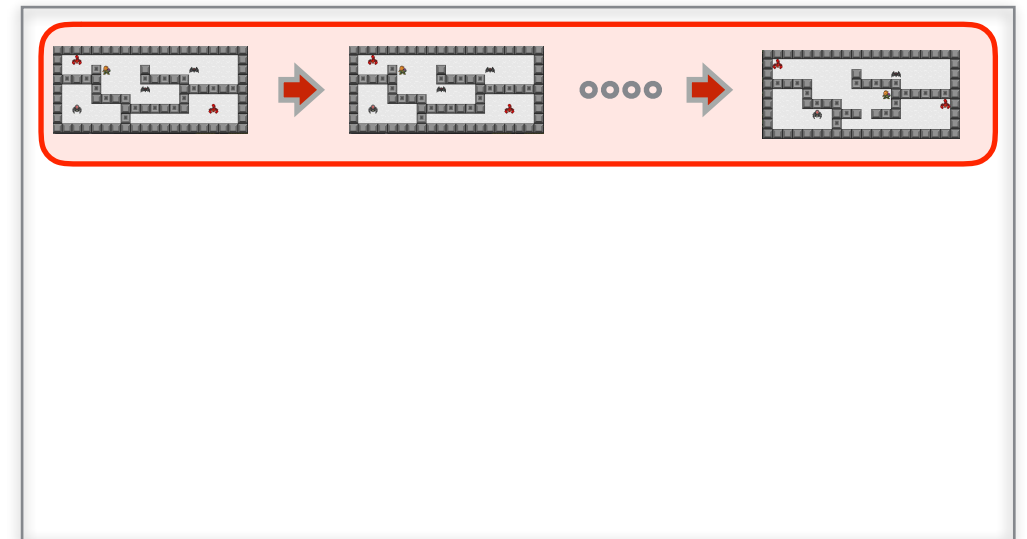
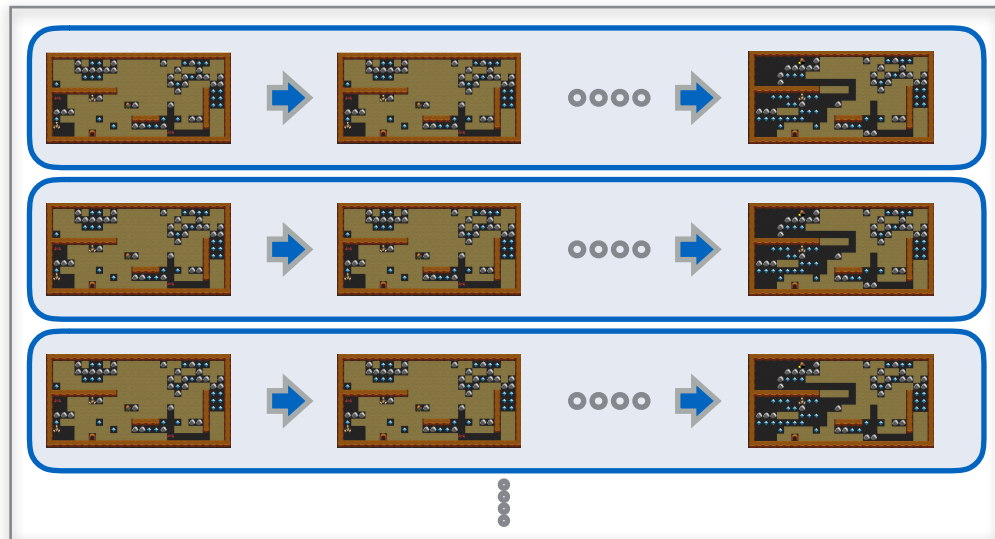


$s'_1$



$s'_n$

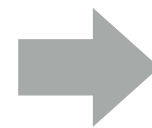
# Bootstrap learning through text



$z_1$

Scorpions chase you  
and kill you on touch

*Transfer*



$z_2$

Spiders are chasers  
and can be destroyed

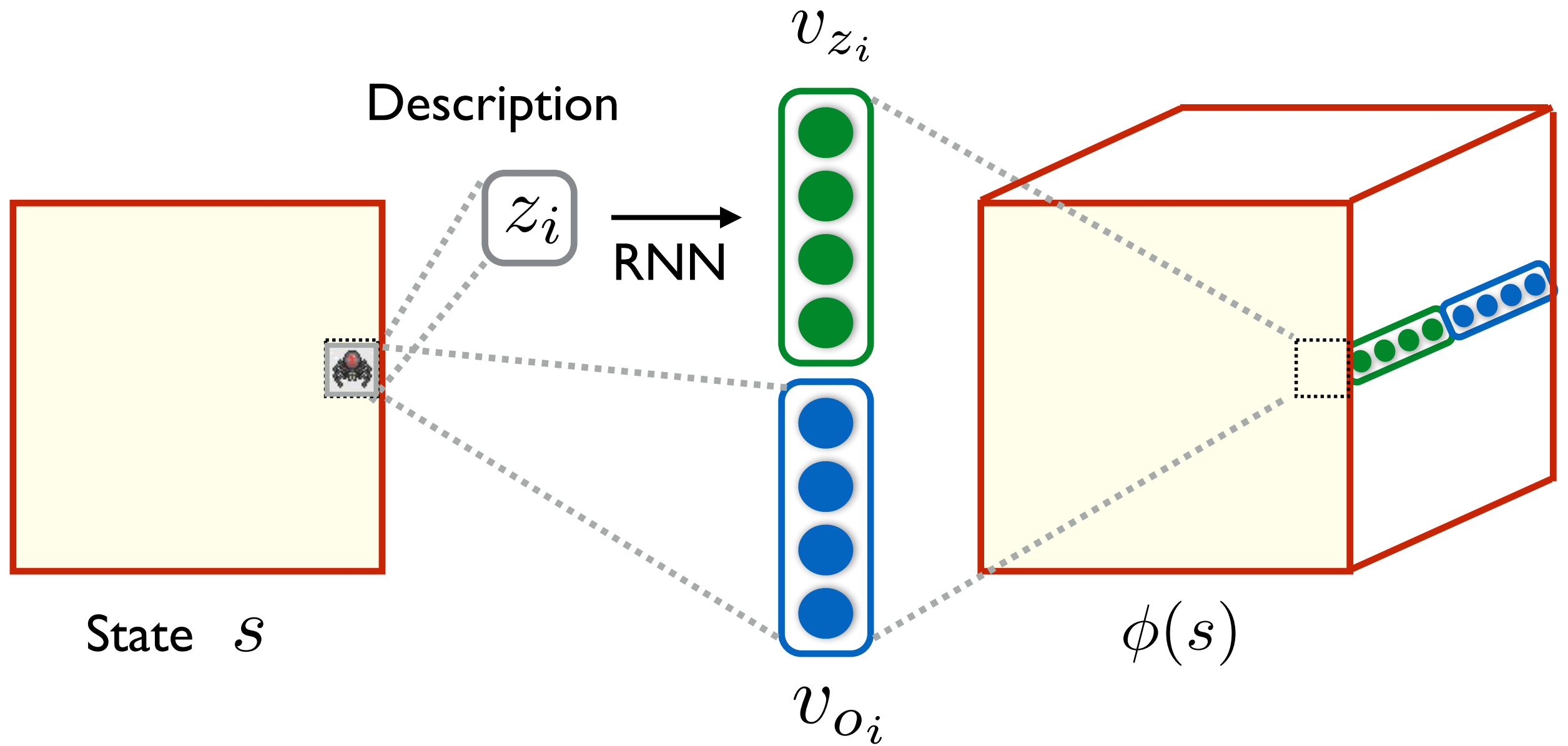
$$T_1(s'|s, a, z_1) \quad R_1(s, a, z_1)$$

$$\hat{T}_2(u'|u, a, z_2) \quad \hat{R}_2(u, a, z_2)$$

- Appropriate representation to incorporate language
- Partial text descriptions

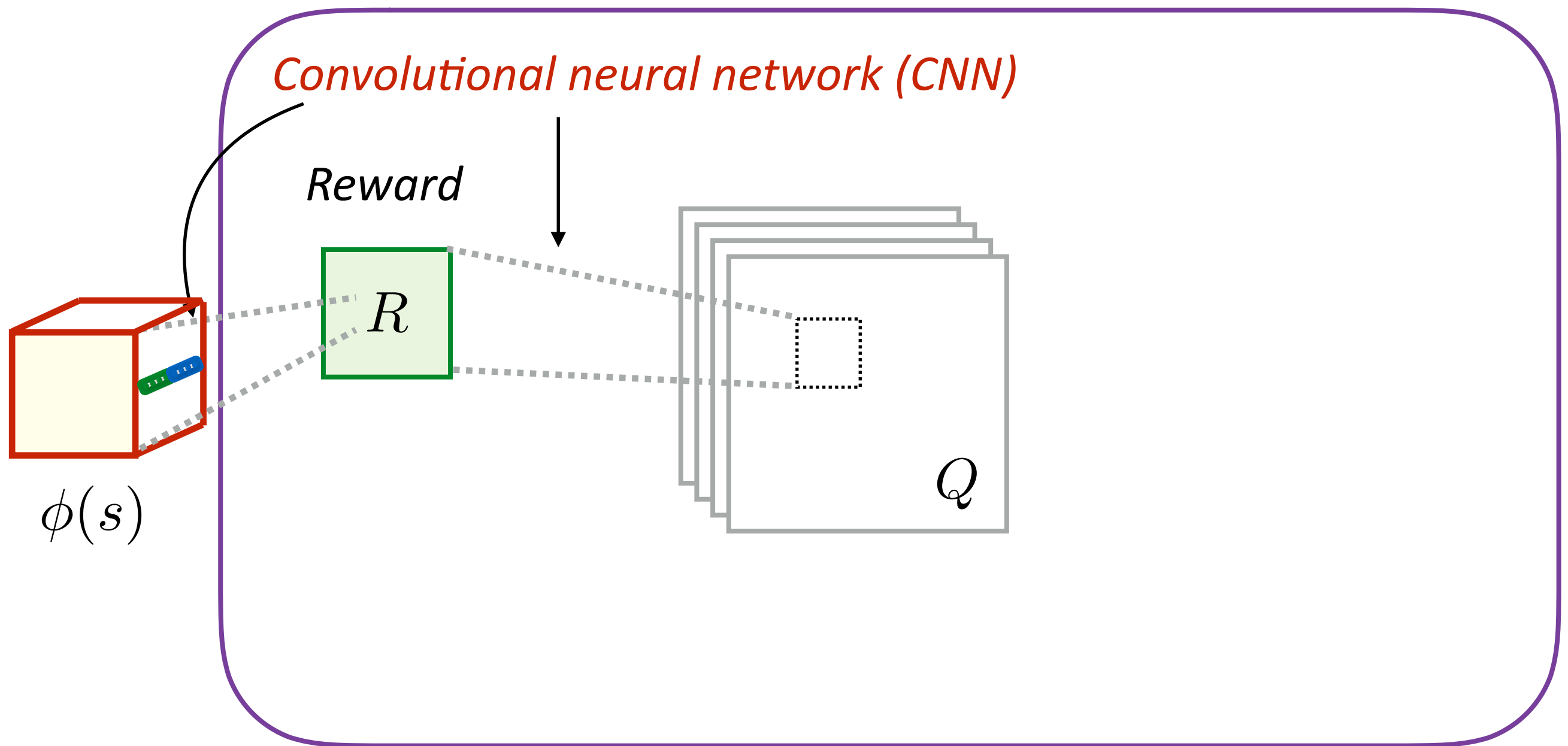


# Incorporating descriptions



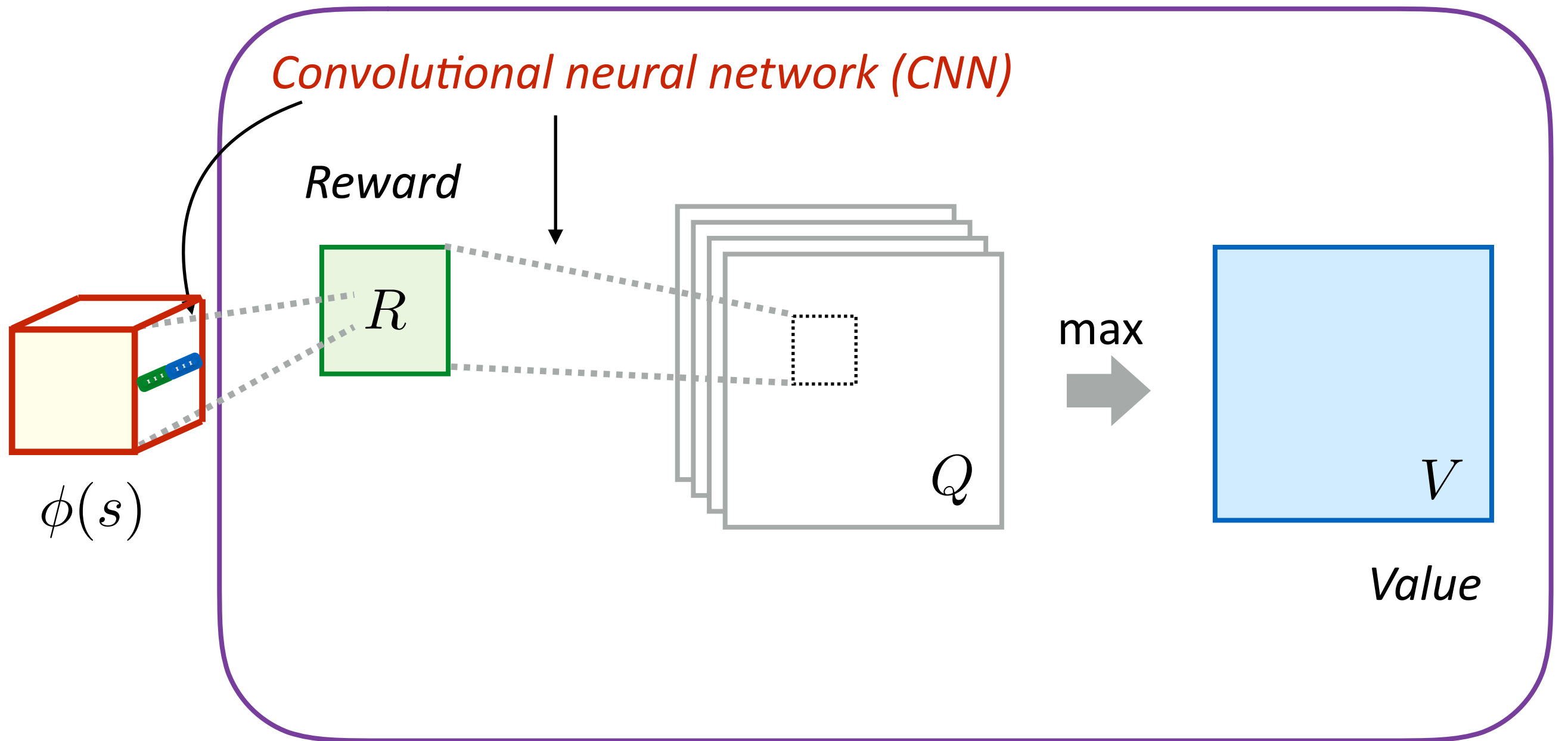
# Differentiable value iteration

$$Q(s, a) = R(s, a) + \gamma \sum_{s'} T(s'|s, a) V(s')$$



# Differentiable value iteration

$$V(s) = \max_a Q(s, a)$$

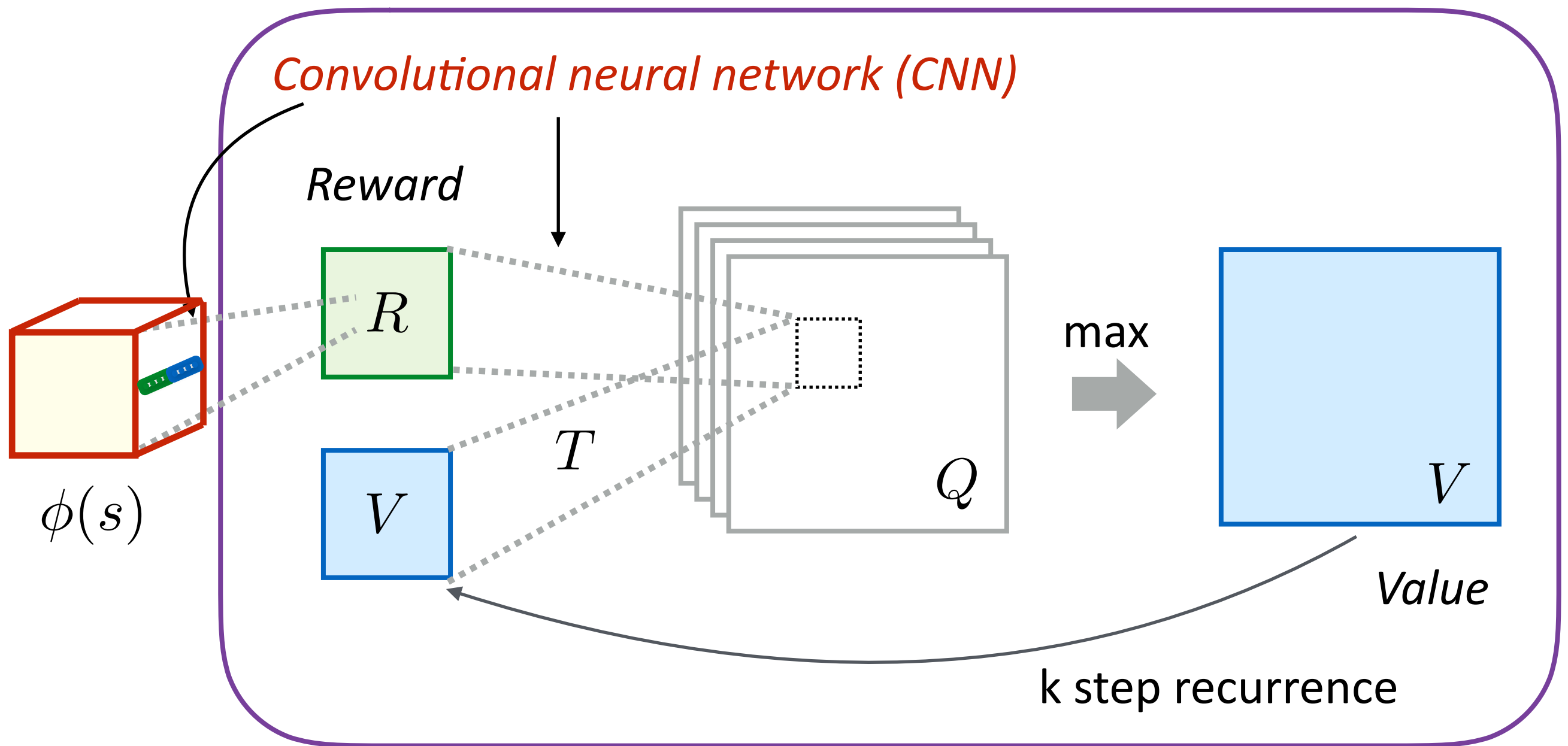


(Value Iteration Network, Tamar et al., 2016)

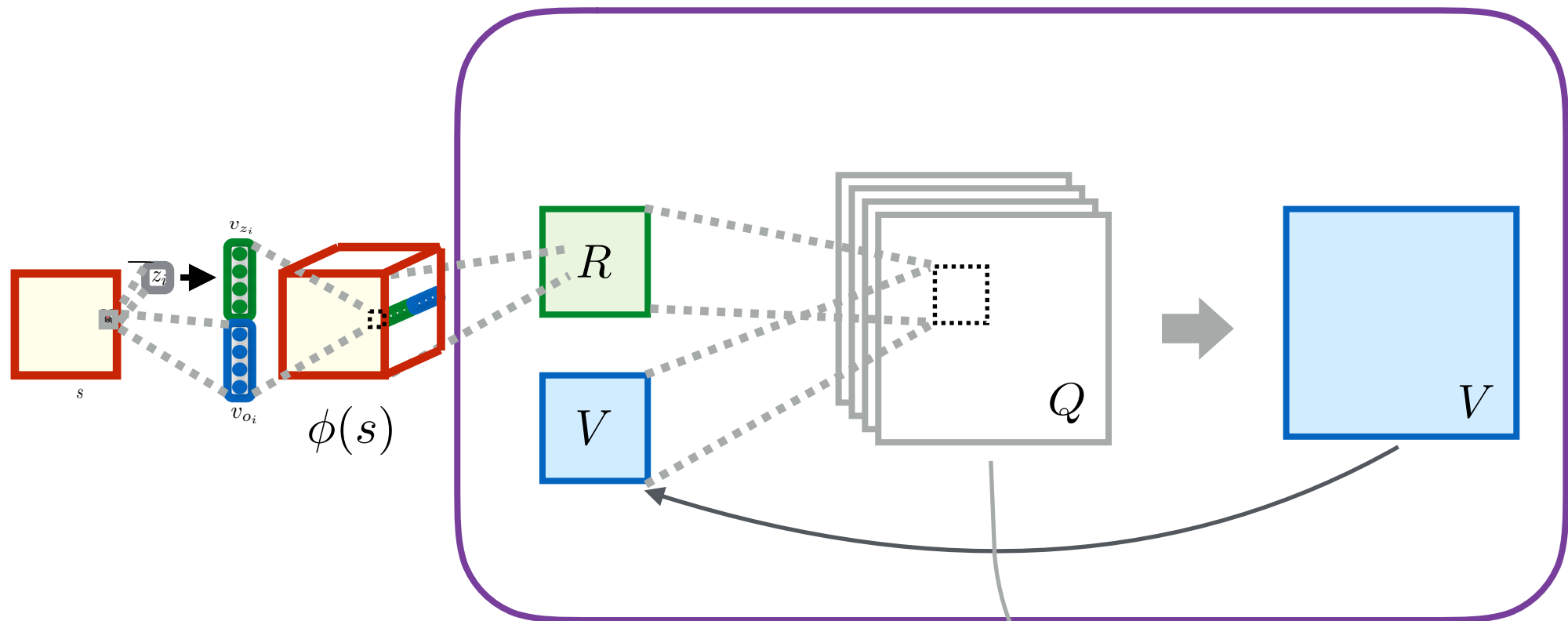
# Differentiable value iteration

$$Q(s, a) = R(s, a) + \gamma \sum_{s'} T(s'|s, a) V(s')$$

$$V(s) = \max_a Q(s, a)$$



# Parameter Learning

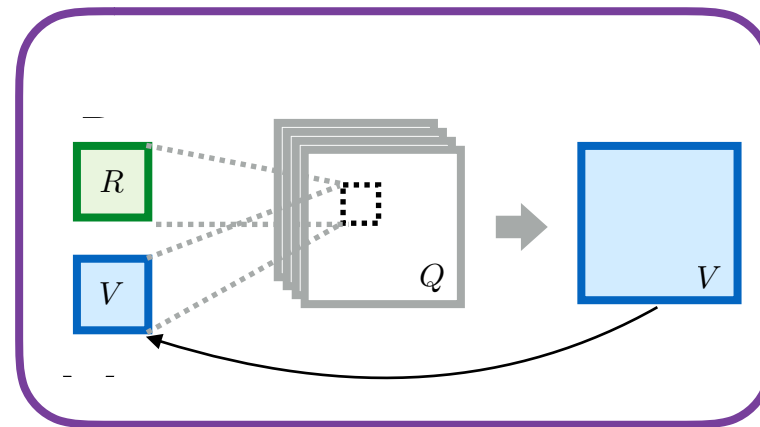
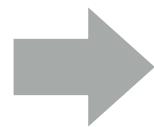


**Objective:** Minimize loss function

$$\mathcal{L}(\Theta_i) = \mathbb{E}_{\hat{s}, \hat{a}, \hat{s}'} [(r + \gamma \max_{a'} Q(\hat{s}', a', Z; \Theta_{i-1}) - Q(\hat{s}, \hat{a}, Z; \Theta_i))^2]$$

# Model-aware policy

Scorpions  
chase you



*Value estimation*



$\pi(a|s)$   
*Policy*



# Model-aware transfer

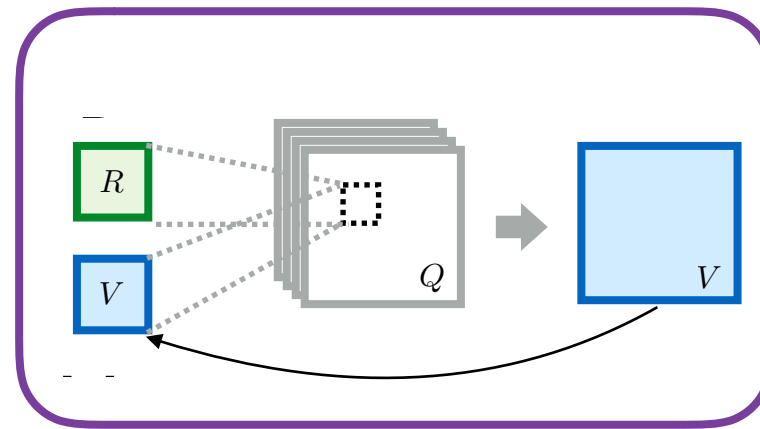
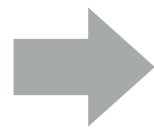
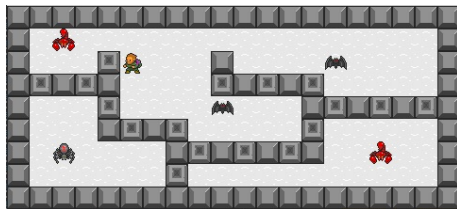
Scorpions  
chase you



Scorpions  
chase you



Spiders are  
chasers



*Value estimation*



$\pi_1(a|s)$

$\pi_2(a|s)$



$\pi_n(a|s)$

*Policies*

# Experiments

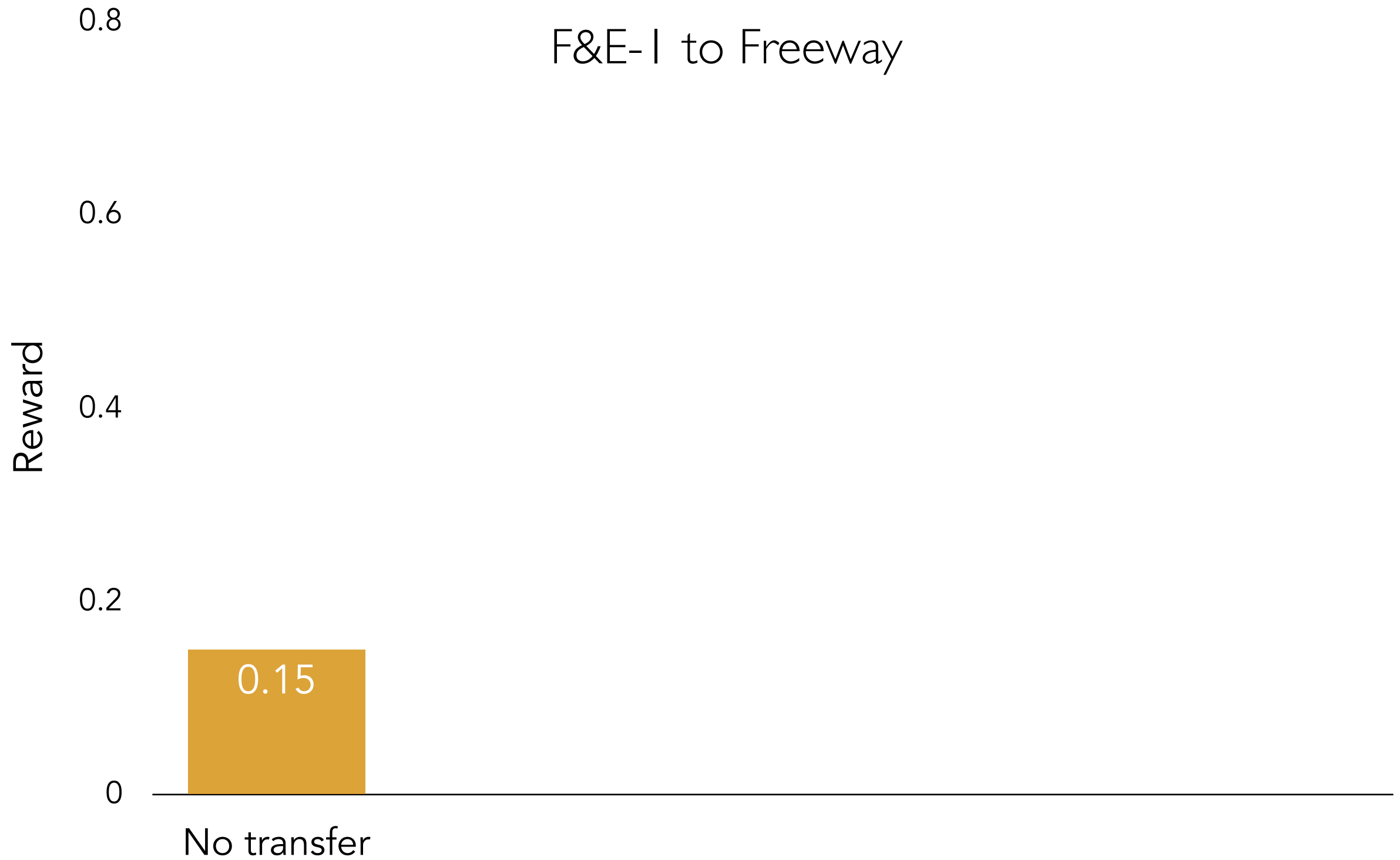
- 2-D game environments with multiple instances (each with different layouts, different entity sets, etc.)
- Text descriptions from Amazon Mechanical Turk
- *Transfer setup*: train on multiple source tasks, and use learned parameters to initialize for target tasks
- *Baselines*: DQN (Mnih et al., 2015), text-DQN, Actor-Mimic (Parisotto et al., 2016)
- *Evaluation*: Jumpstart, average and asymptotic reward

Condition	Source	Target
F&E-1 → F&E-2	7	3
F&E-1 → Freeway	7	5
Bomberman → Boulderchase	5	5

Source and target game instances for transfer

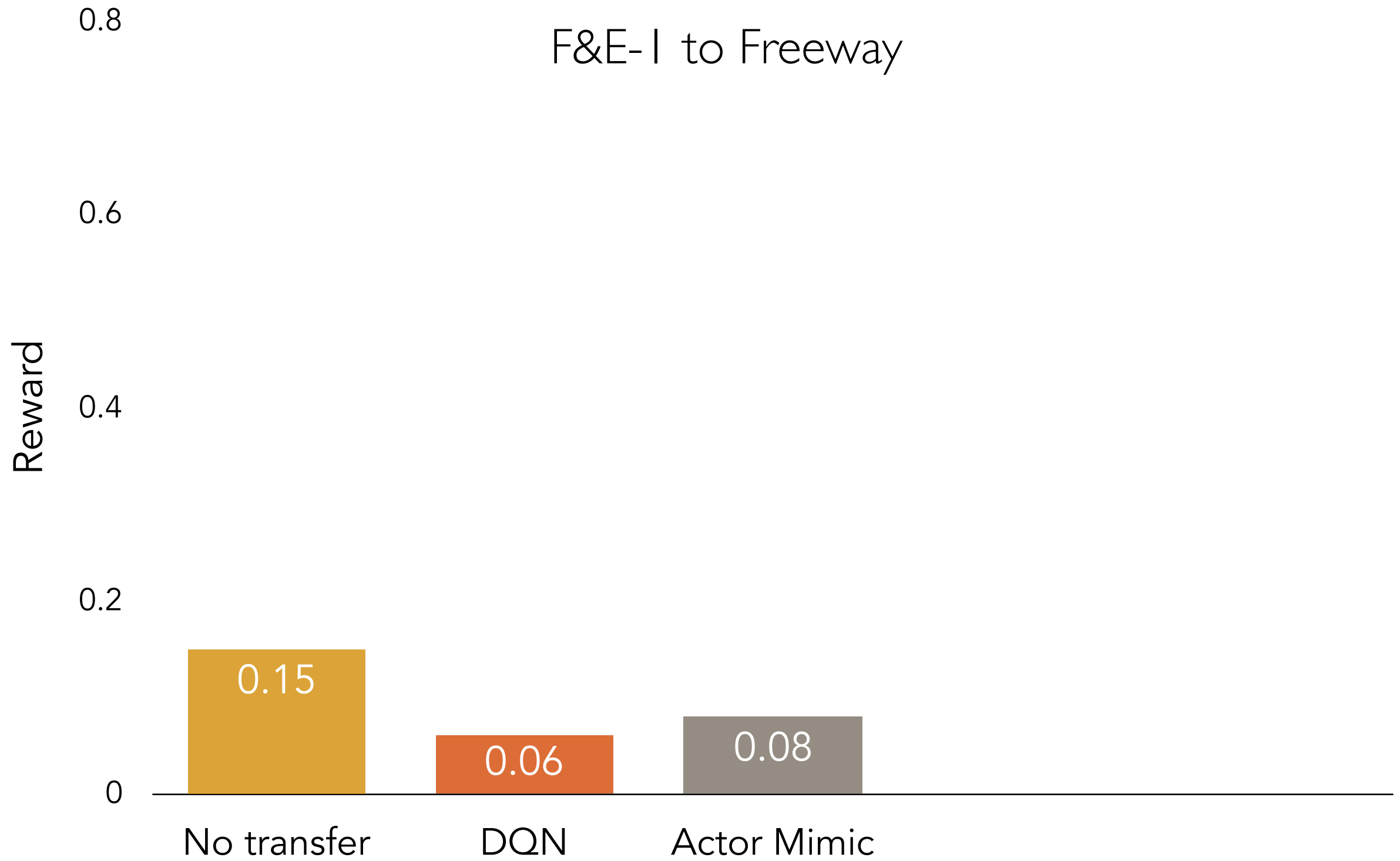
# Average reward

F&E-I to Freeway



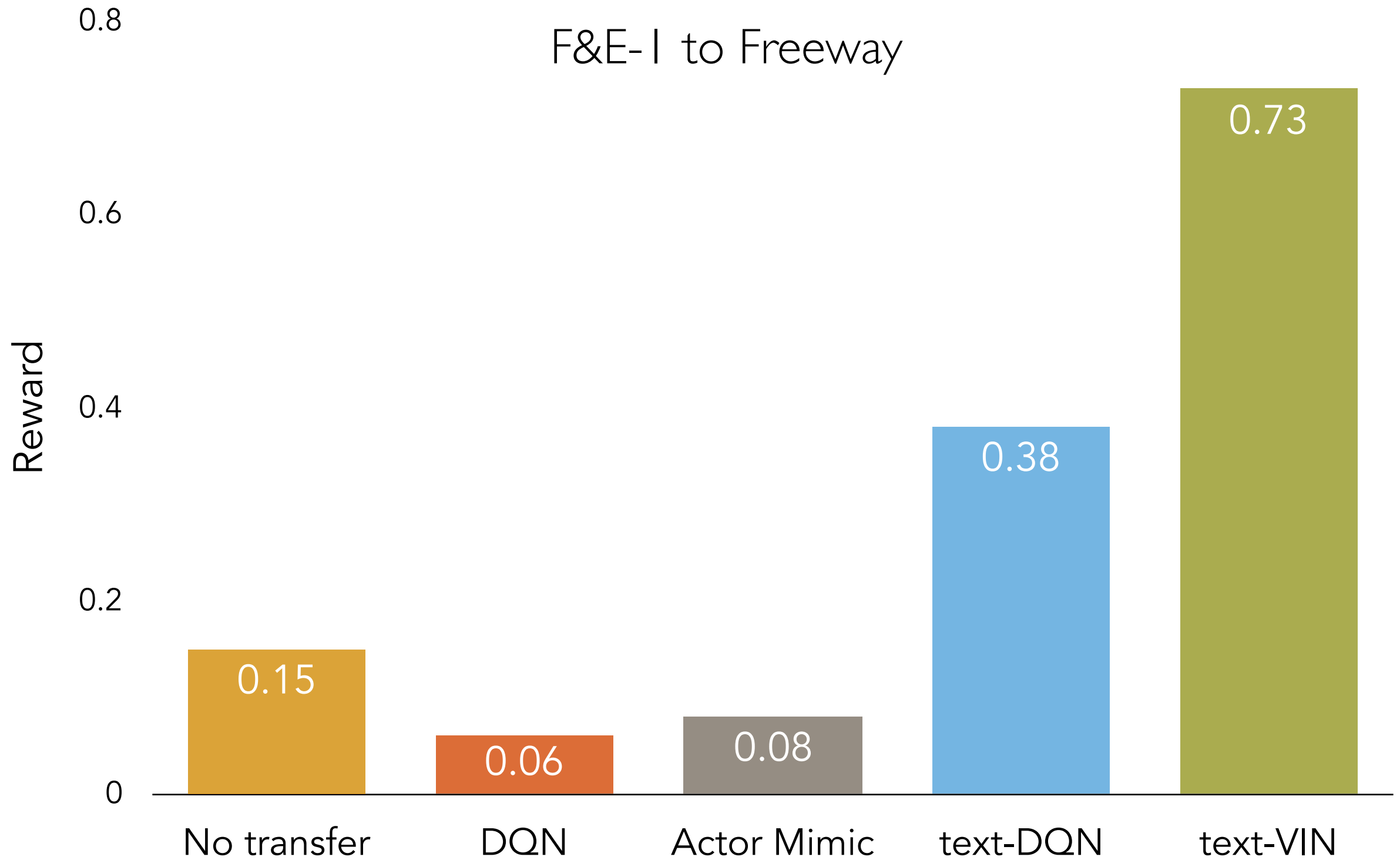
# Average reward

F&E-I to Freeway

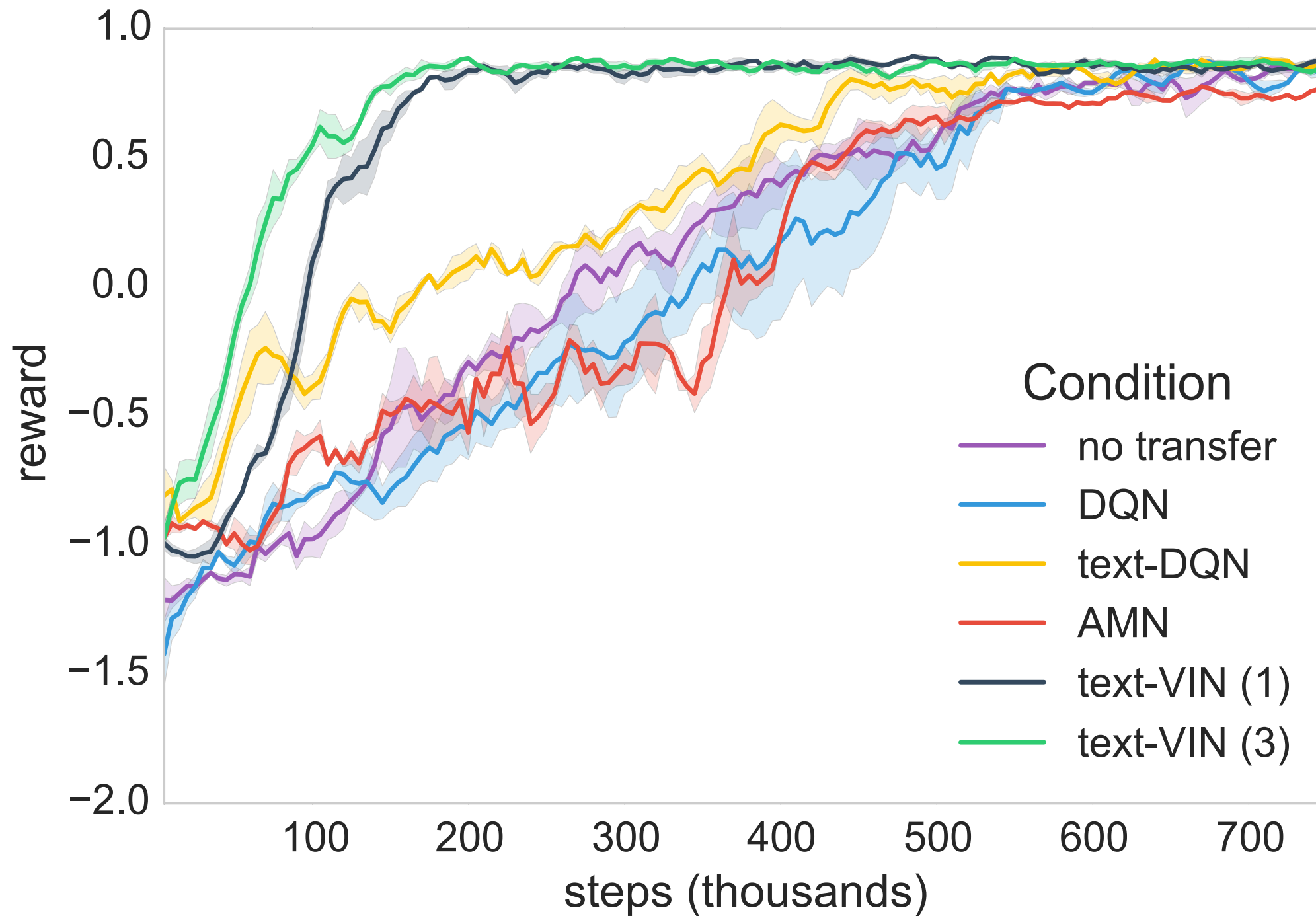


# Average reward

F&E-I to Freeway

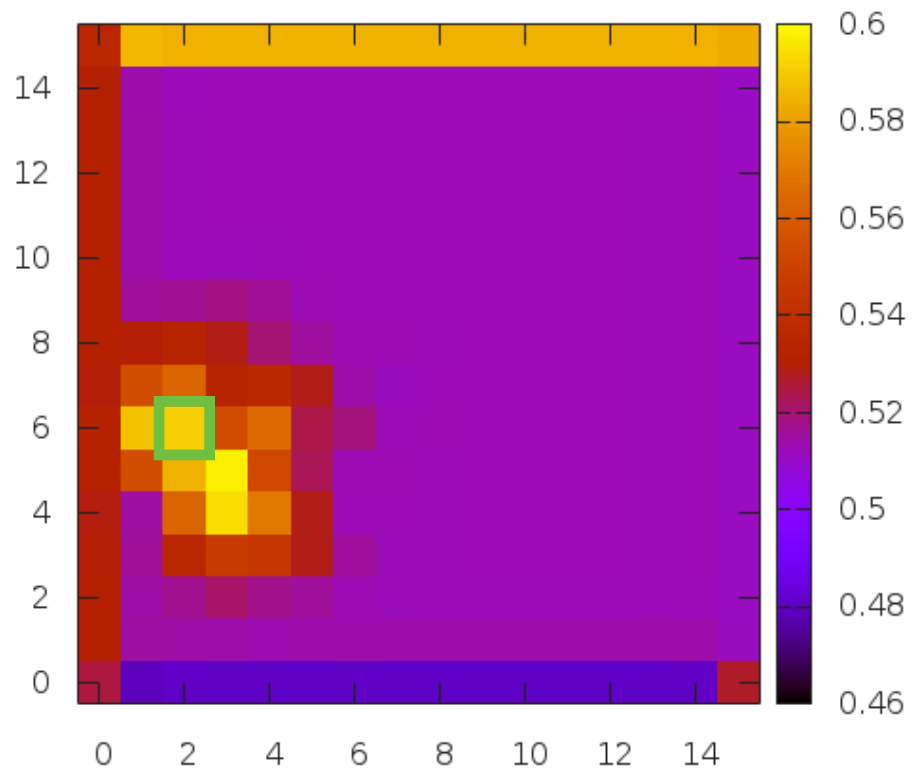


# Transfer results (F&E-I to Freeway)

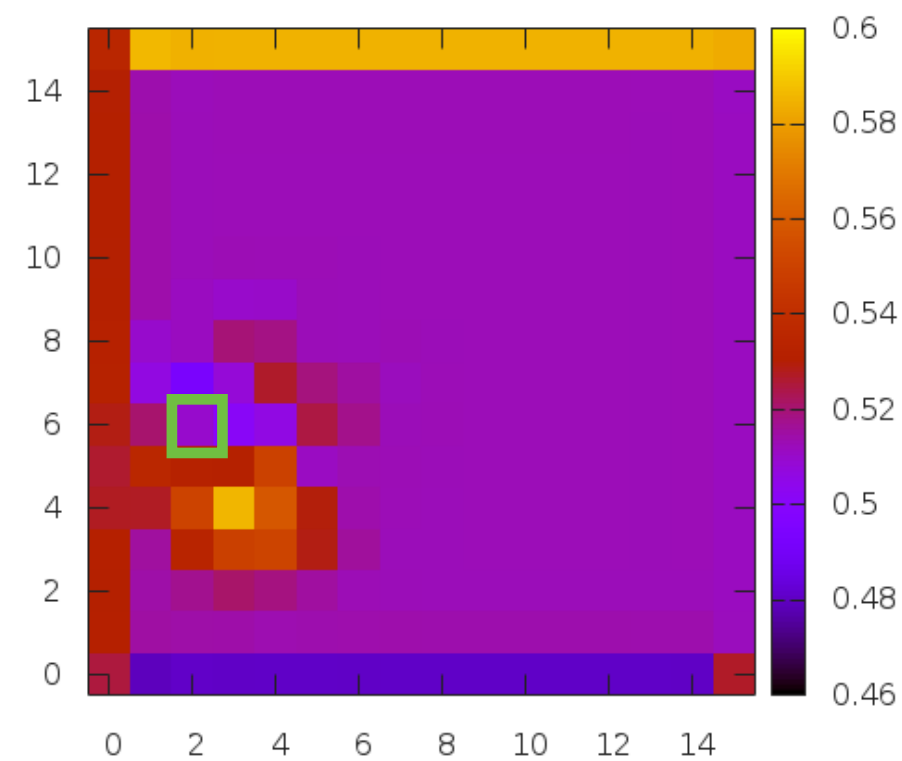


Agent: (3,4)

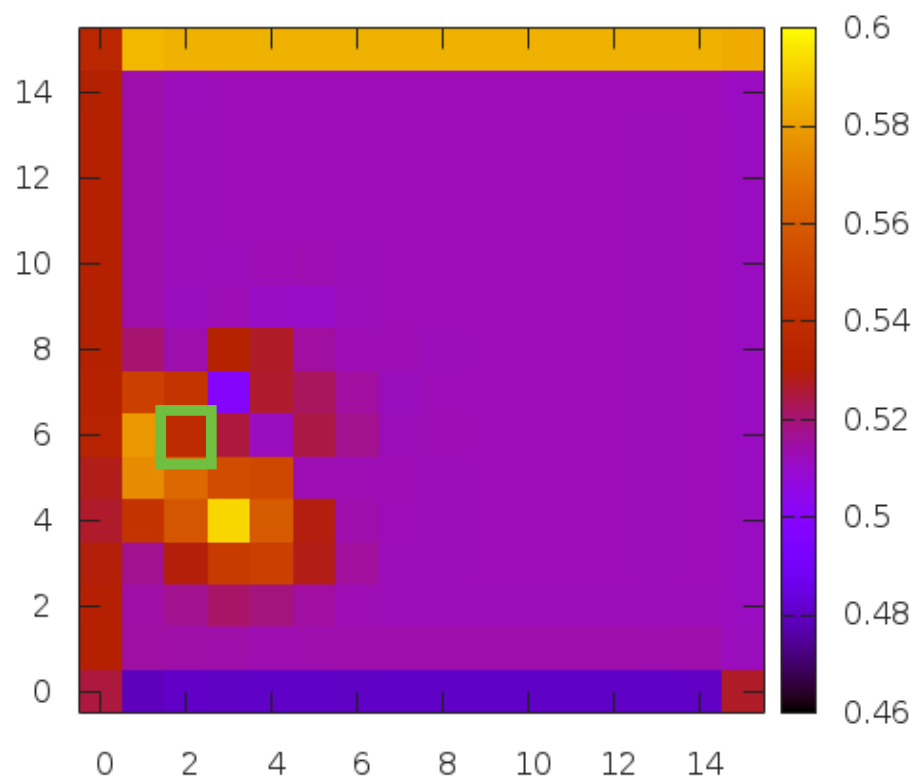
Entity: (2,6)



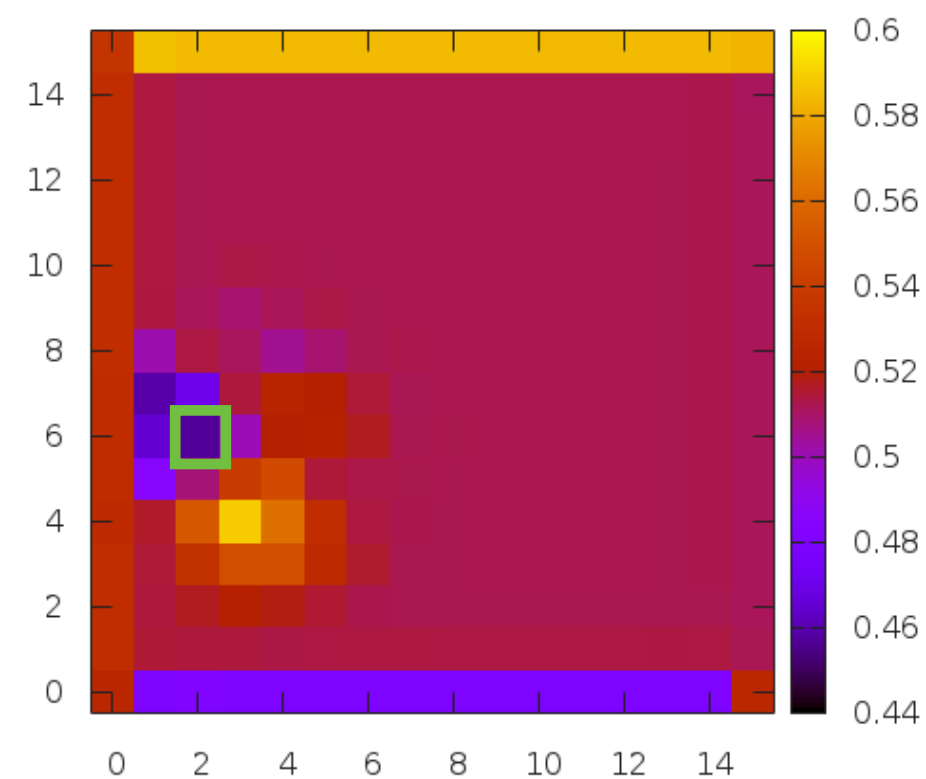
Seen 'friendly' entity



Unseen entity



Unseen entity + 'friendly' text



Unseen entity + 'enemy' text

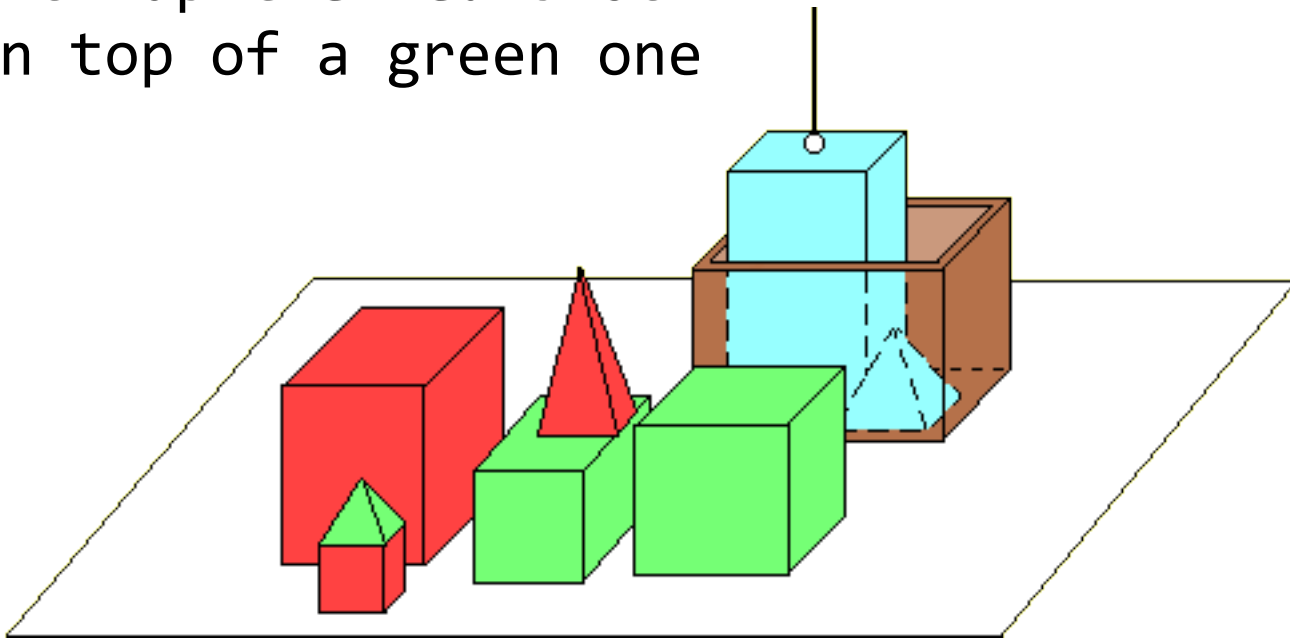
# Representation Learning for Grounded Spatial Reasoning

*Michael Janner, Karthik Narasimhan, Regina Barzilay*  
*MIT*



# Understanding spatial references

Pick up the red block  
on top of a green one

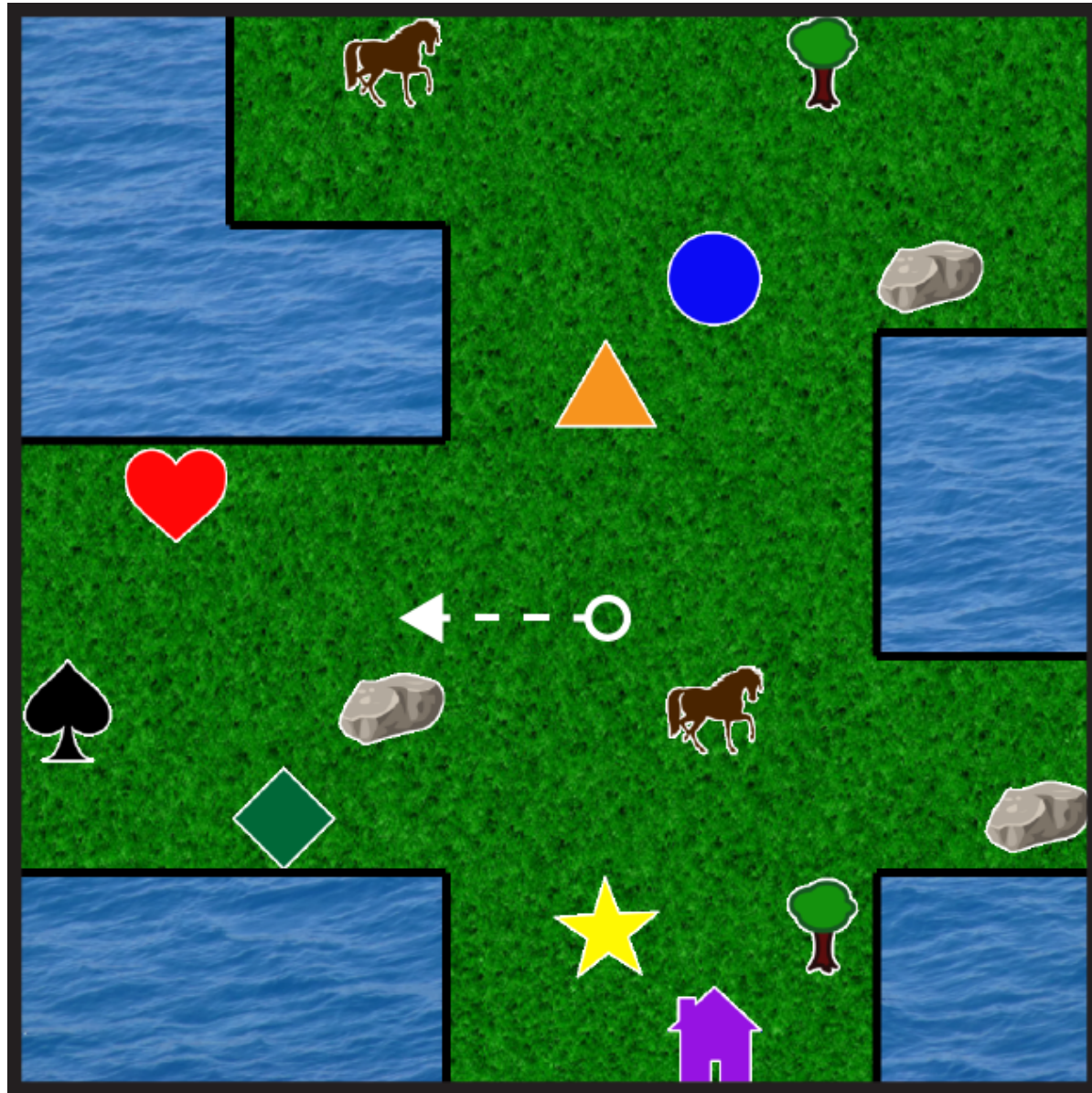


Human robot interaction



Autonomous navigation

# A spatial reasoning task

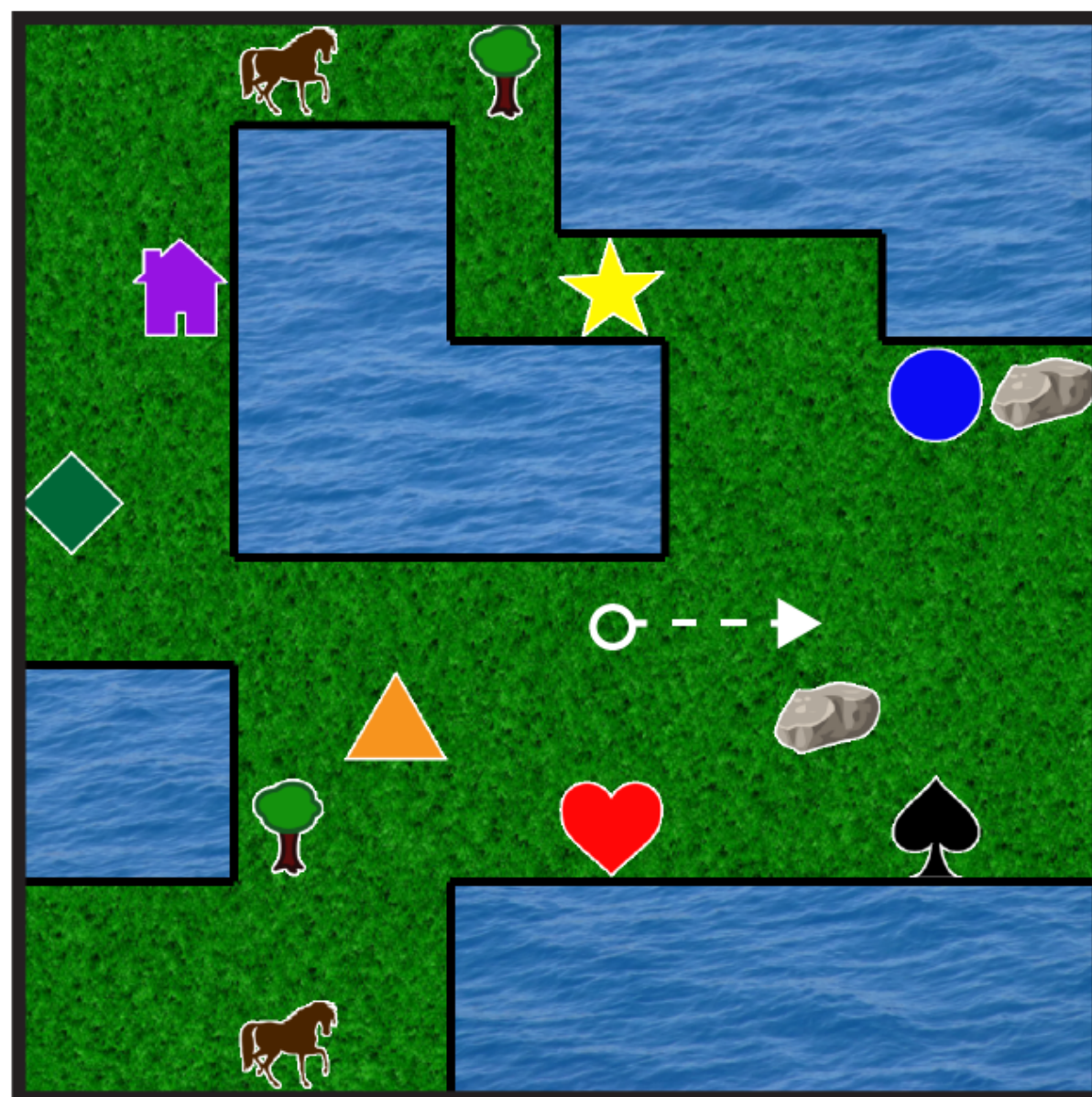
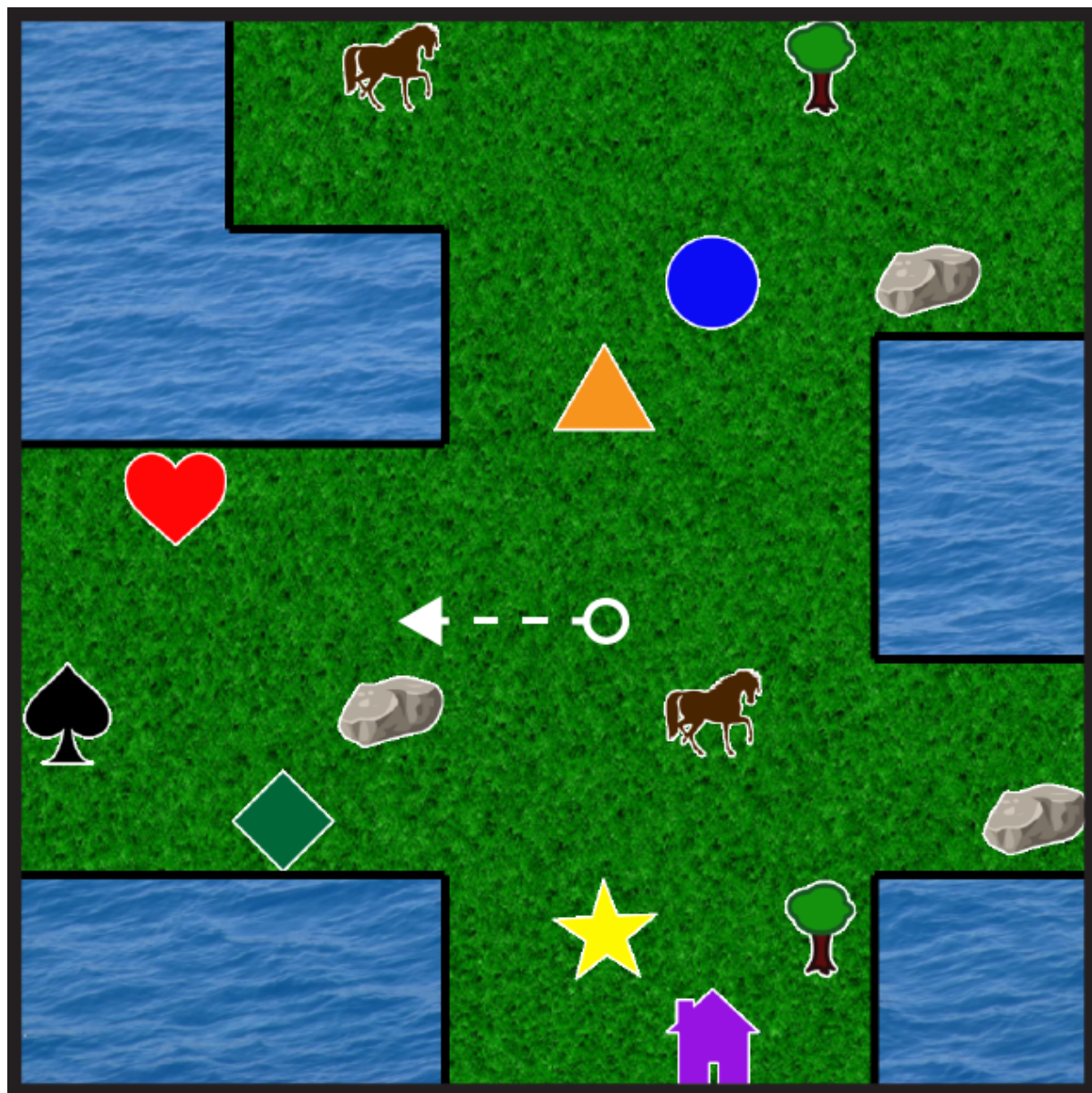


- Interactive navigation world
- Goal specified in natural language
- Rewards for reaching goals
- No domain knowledge

*Reach the cell above the westernmost rock*



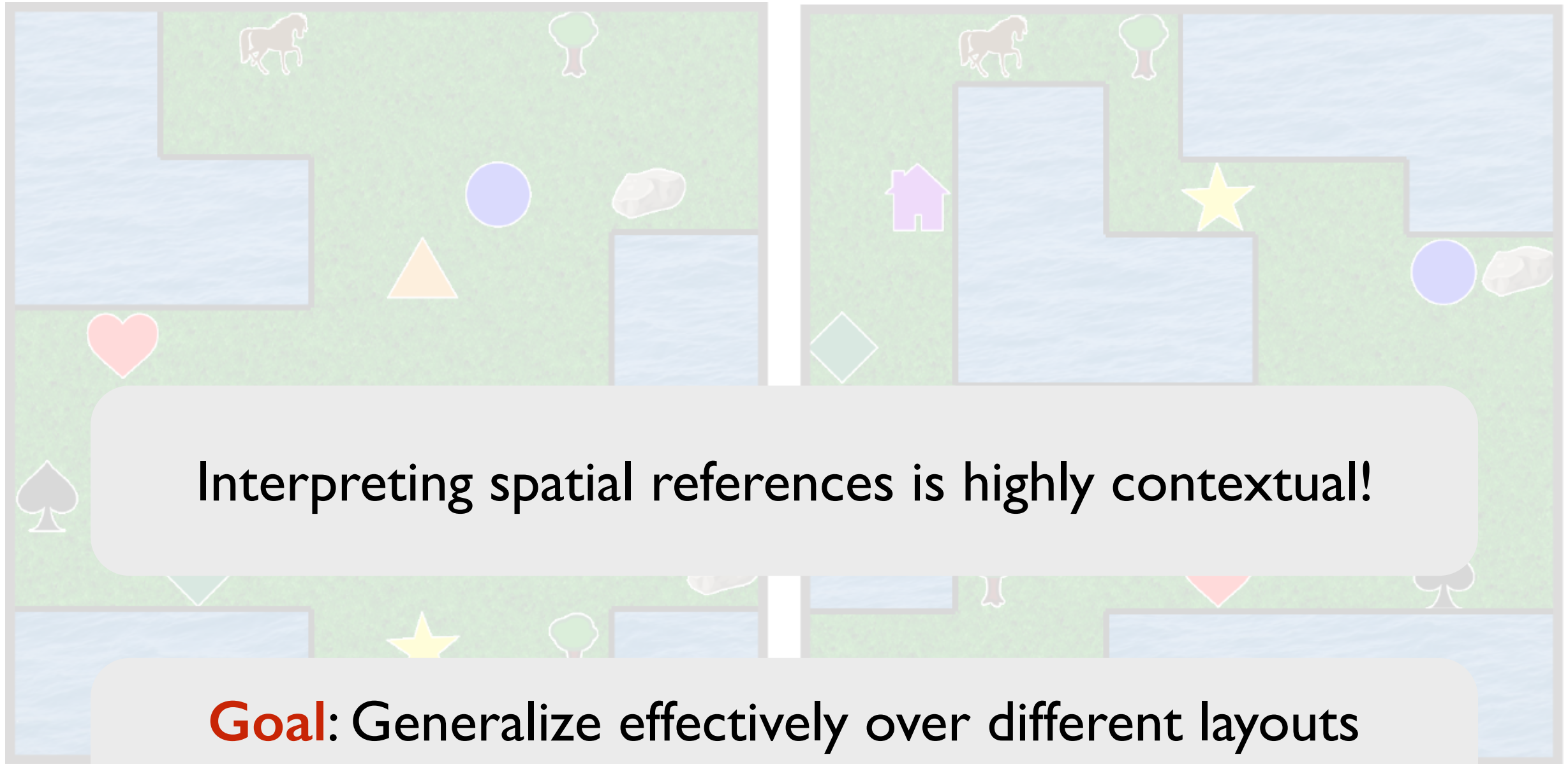
# A spatial reasoning task



*Reach the cell above the westernmost rock*



# A spatial reasoning task



Interpreting spatial references is highly contextual!

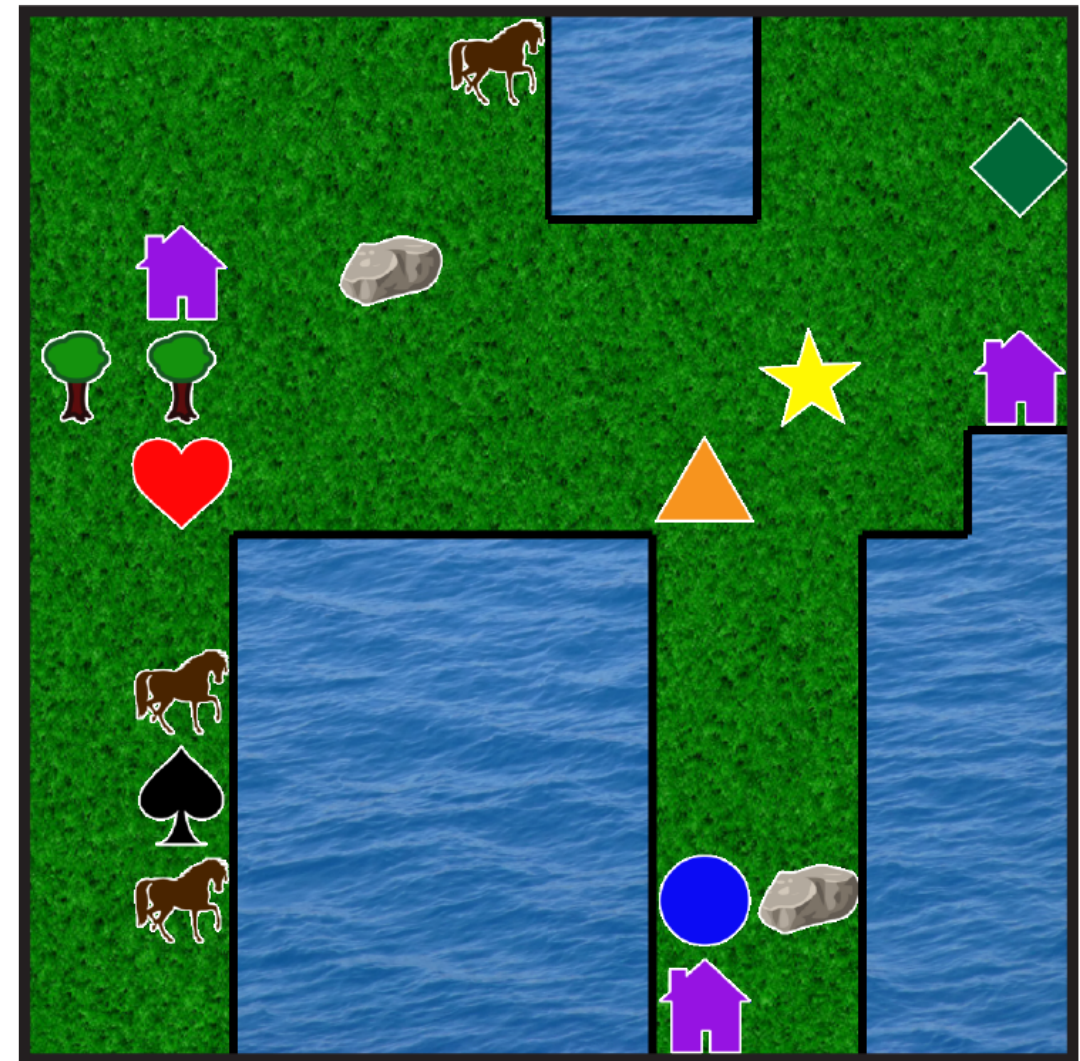
**Goal:** Generalize effectively over different layouts and spatial references

*Reach the cell above the westernmost rock*

# Types of spatial references

I. Refer to specific entity

*“Go to the circle”*

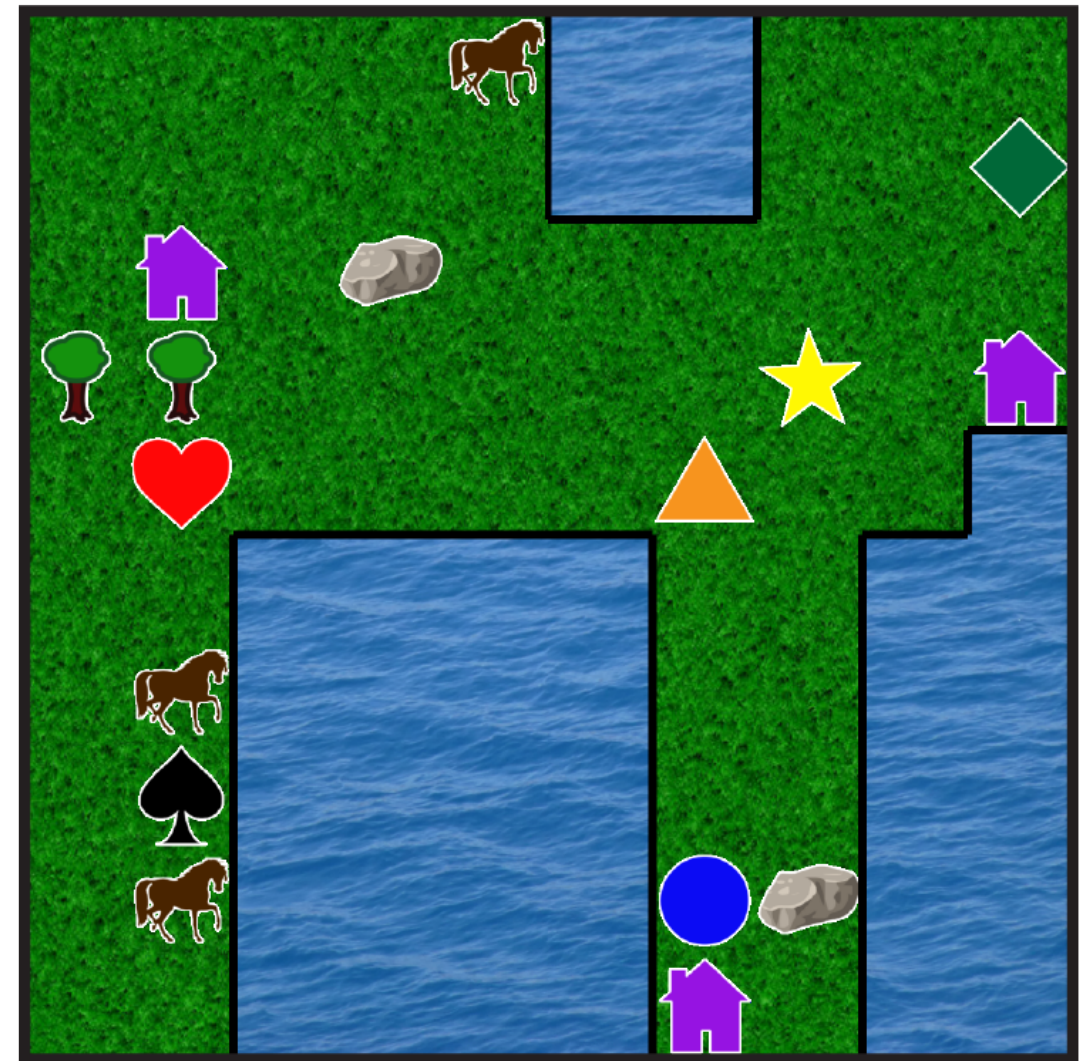




# Types of spatial references

1. Refer to specific entity
2. Location using a single referent entity

*“Reach the cell above the circle”*

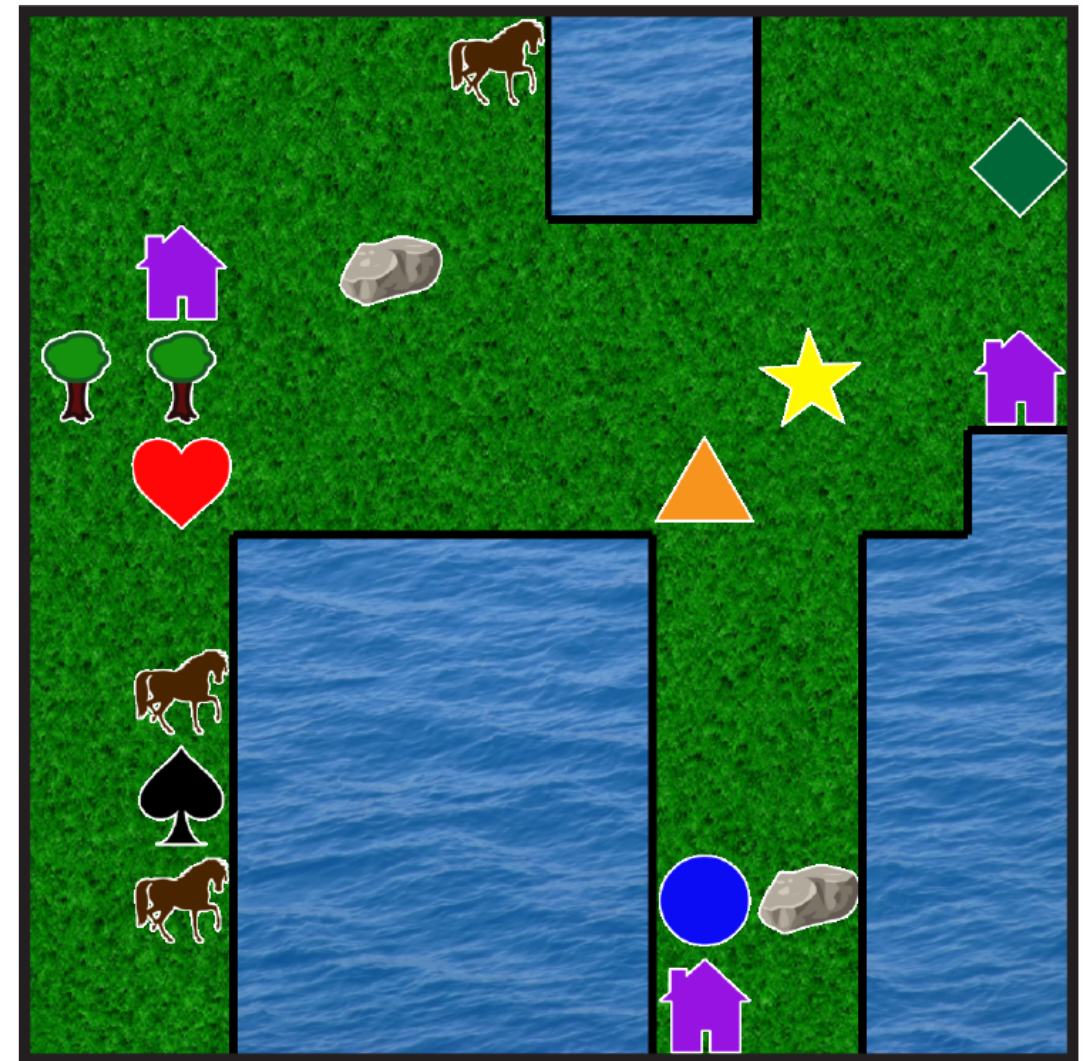




# Types of spatial references

1. Refer to specific entity
2. Location using a single referent entity
3. Location using multiple referent entities

*“Move to the goal one square to the right of triangle and two squares to the bottom of star”*

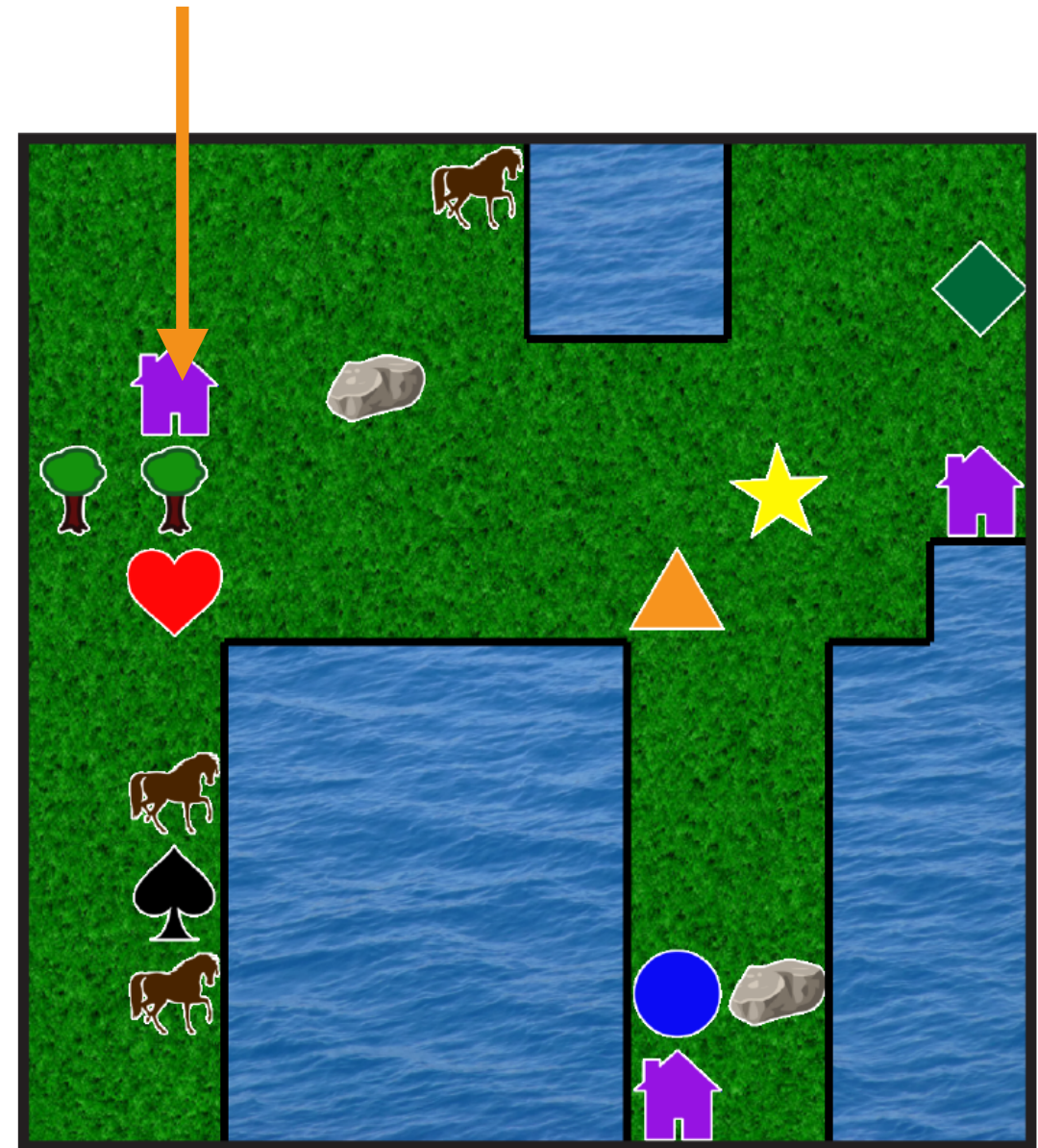


# Types of spatial references

1. Refer to specific entity
2. Location using a single referent entity
3. Location using multiple referent entities



1. *(Local) Go two spaces above the heart.*
2. *(Global) Reach the northernmost house.*



# Challenges

- Interpretation of spatial references is **highly context-dependent**.
- **Rich, flexible ways** of verbalizing spatial references
- Only source of supervision is **reward-based feedback**

# Markov Decision Process

$\langle S, A, X, T, R \rangle$

$S$  : *State configurations*

$A$  : *Actions*

$X$  : *Goal specifications in language*

$T$  : *Transition distribution*

$R$  : *Reward function*

# Value Iteration

$$Q(s, a, x) = R(s, x) + \gamma \sum_{s' \in S} T(s' | s, a, x) V(s', x)$$

$$V(s, x) = \max_a Q(s, a, x)$$

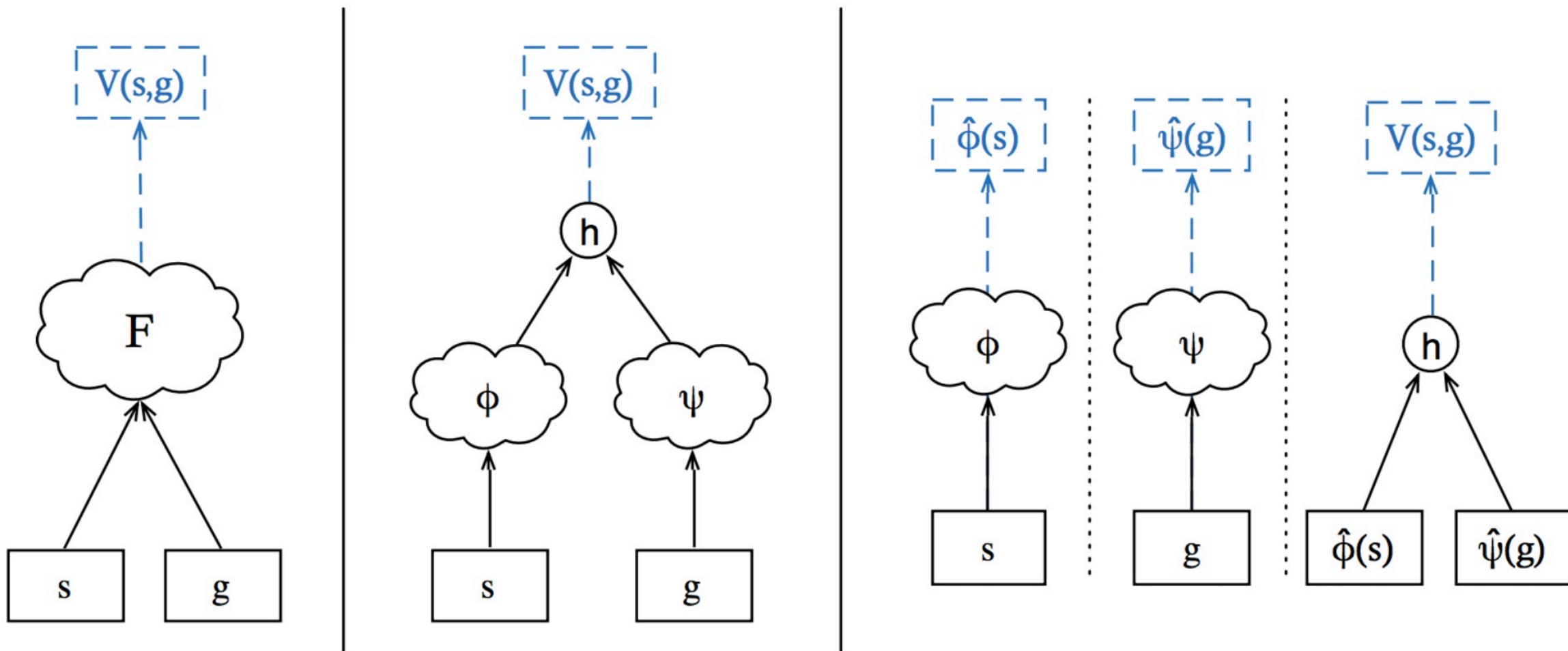
**Goal-conditioned action policy:**  $\pi(s, x) = \arg \max_a Q(s, a, x)$



# Model requirements

- Joint representation of observations (s) and instructions (x)
- Flexible representation of goals, encoding both local structure and global spatial attributes.
- Must be compositional, to generalize over linguistic variety in instructions.

# Universal Value Function Approximators

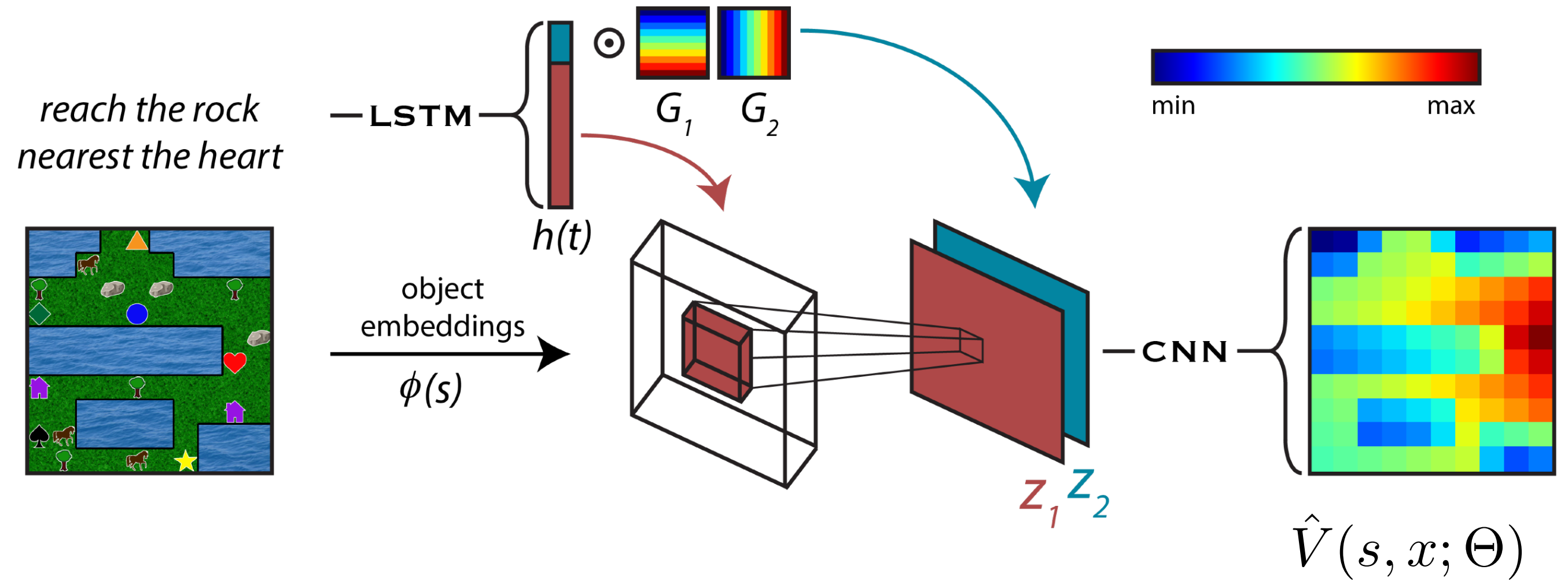


(Schaul et al., 2015)





# Our Model



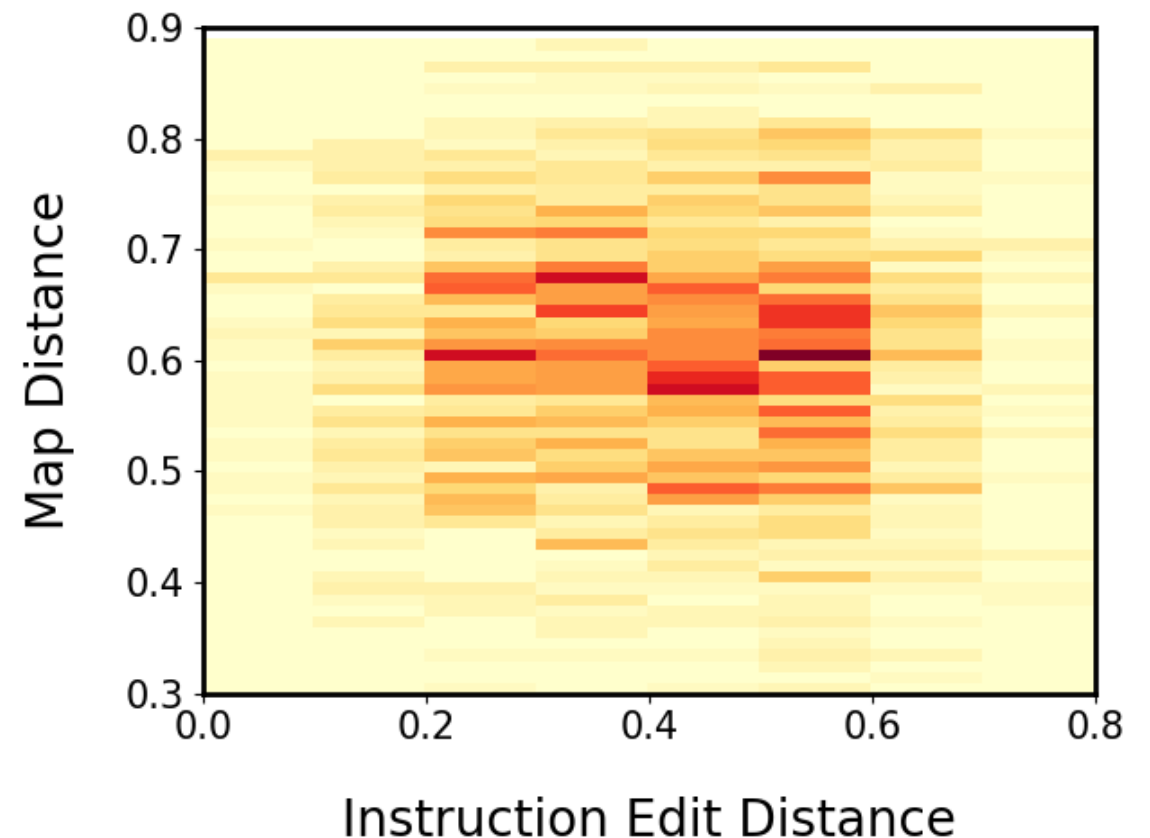
$$\mathcal{L}(\Theta) = \mathbb{E}_{s \sim \mathcal{D}} \left[ \hat{V}(s, x; \Theta) - \left( R(s, x) + \gamma \max_a \sum_{s'} T(s'|s, a) \hat{V}(s', x; \Theta^-) \right) \right]^2$$



# Experimental setup

- Puddle world with randomized layout and randomly placed unique (6) and non-unique (4) objects
- Text instructions for randomized goals collected from Amazon Mechanical Turk (max length 43)

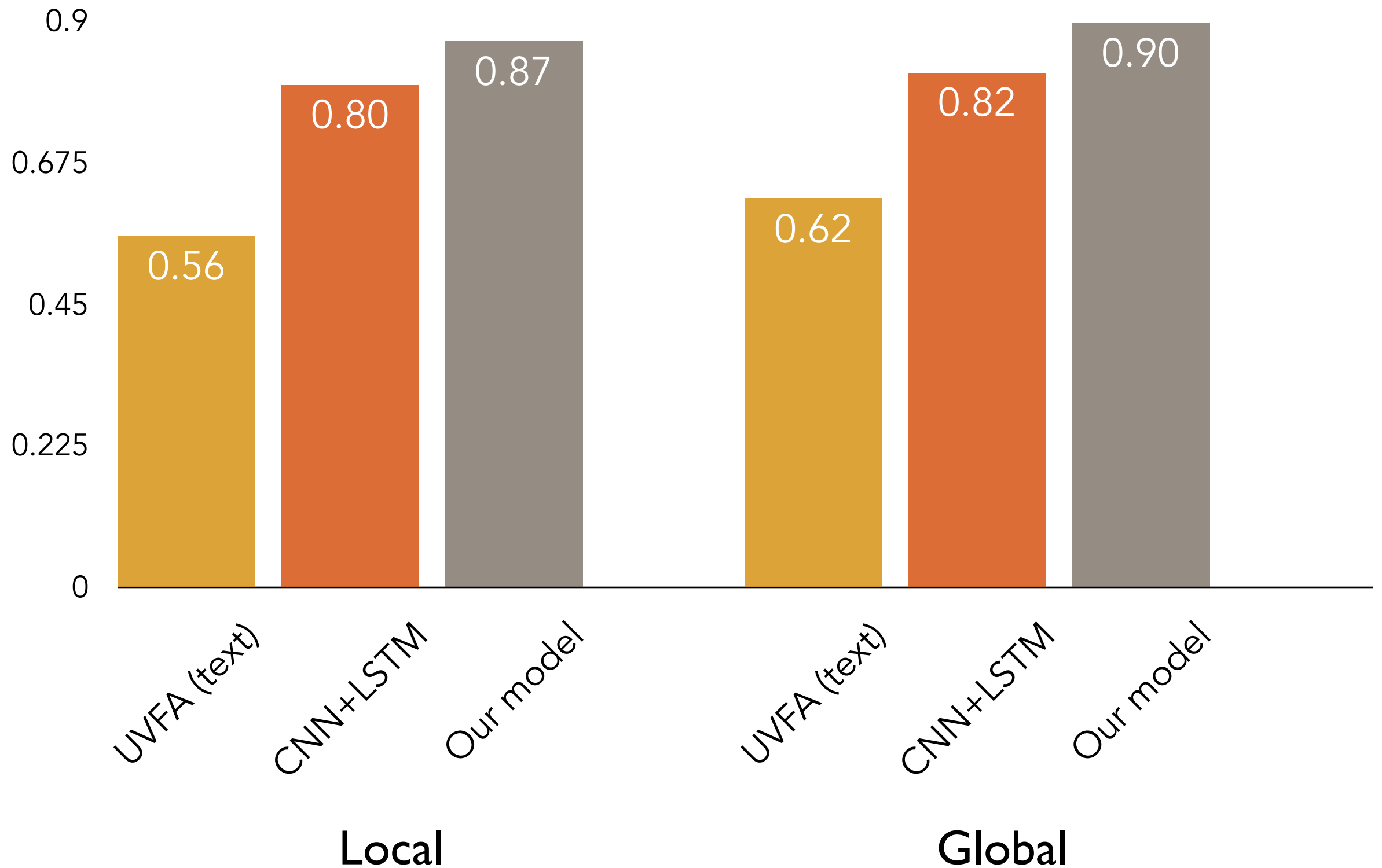
Split	Local	Global
Train	1566	1071
Test	399	272



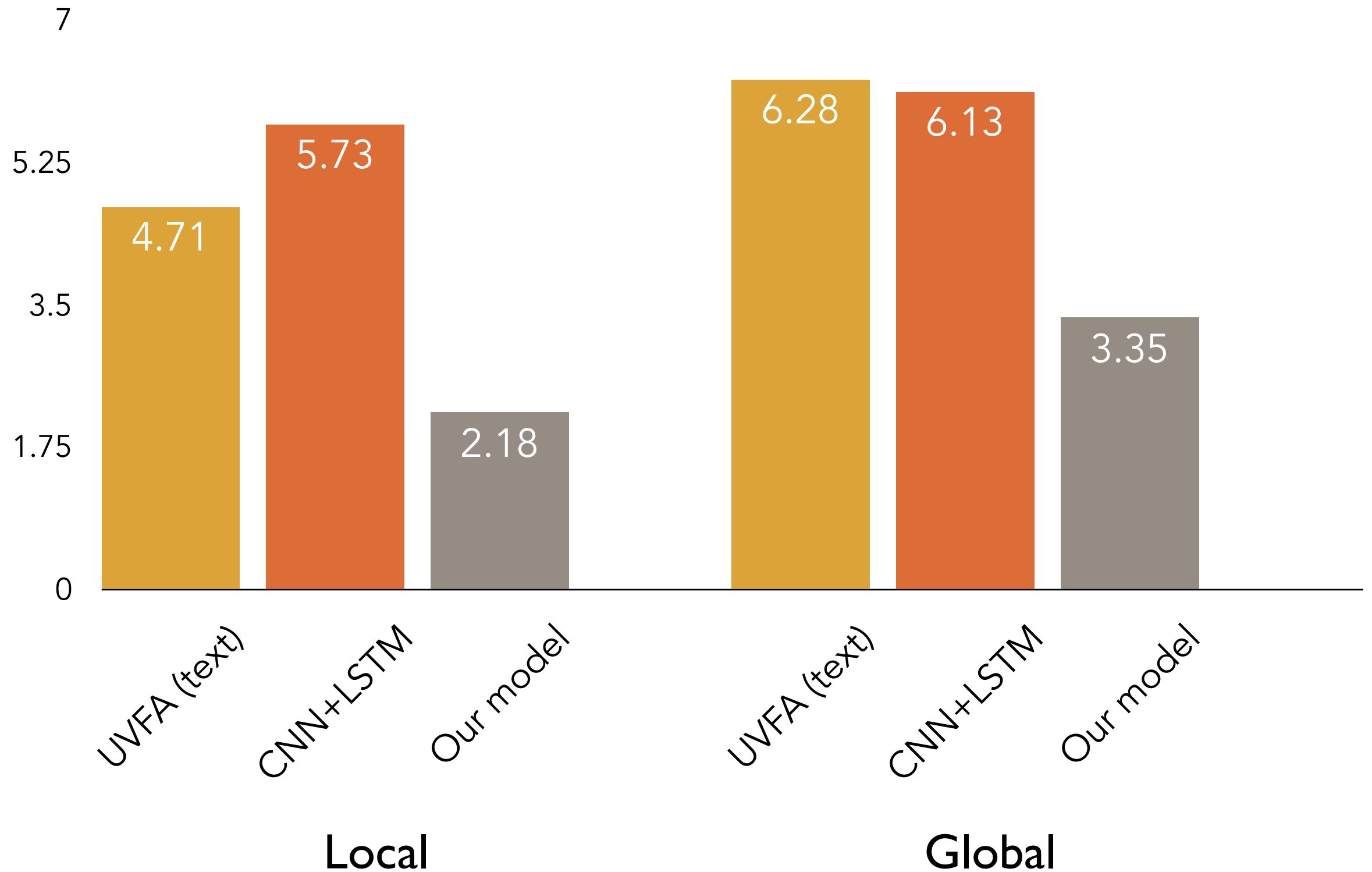
# Baselines

- UVFA (text): UVFA model (Schaul et al., 2015) adapted to use text for goal specifications
- CNN+LSTM: Separately process image and text and then feed concatenated representations to MLP (Misra et al., 2017)

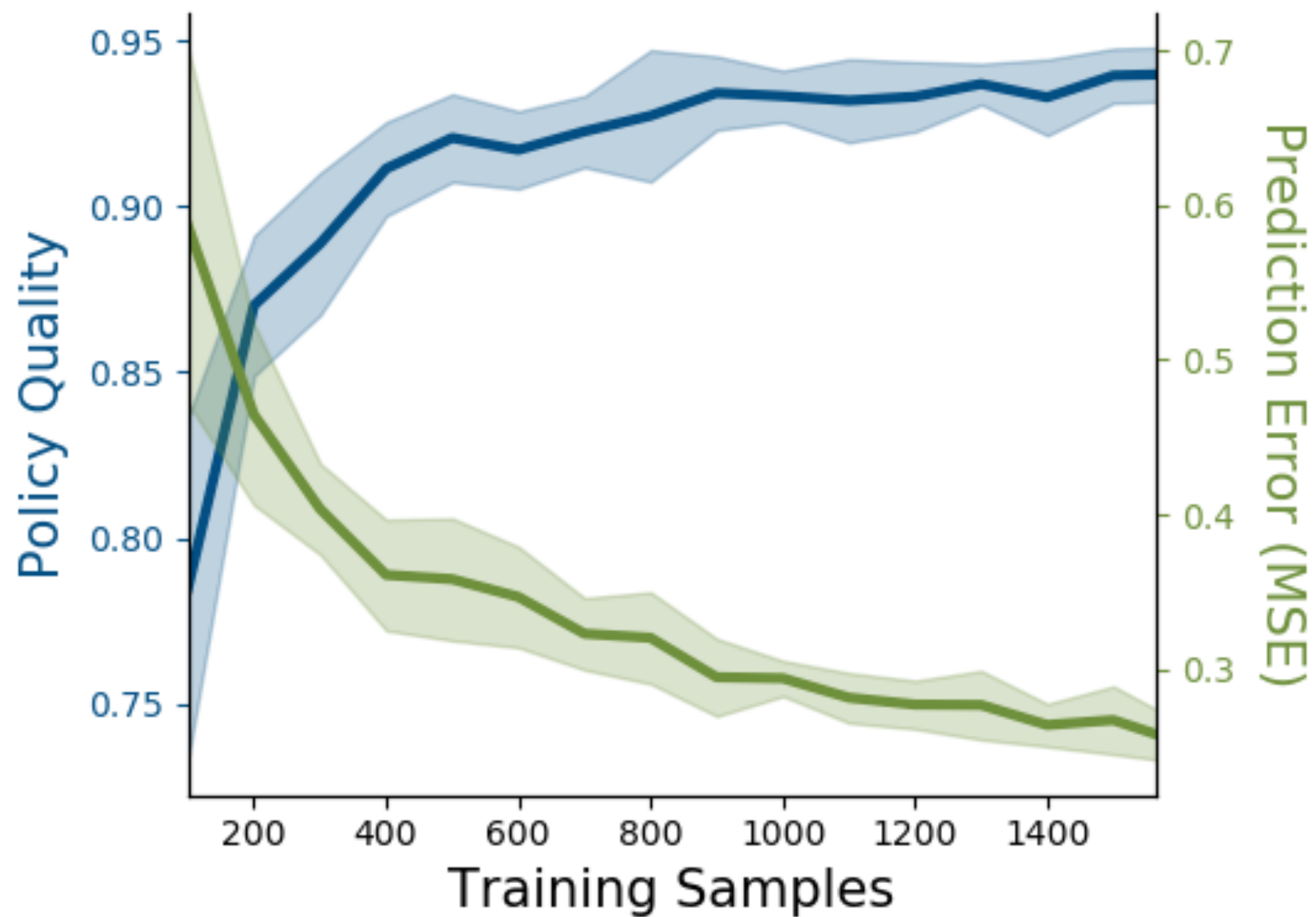
# Results: Policy Quality



# Results: Distance to goal

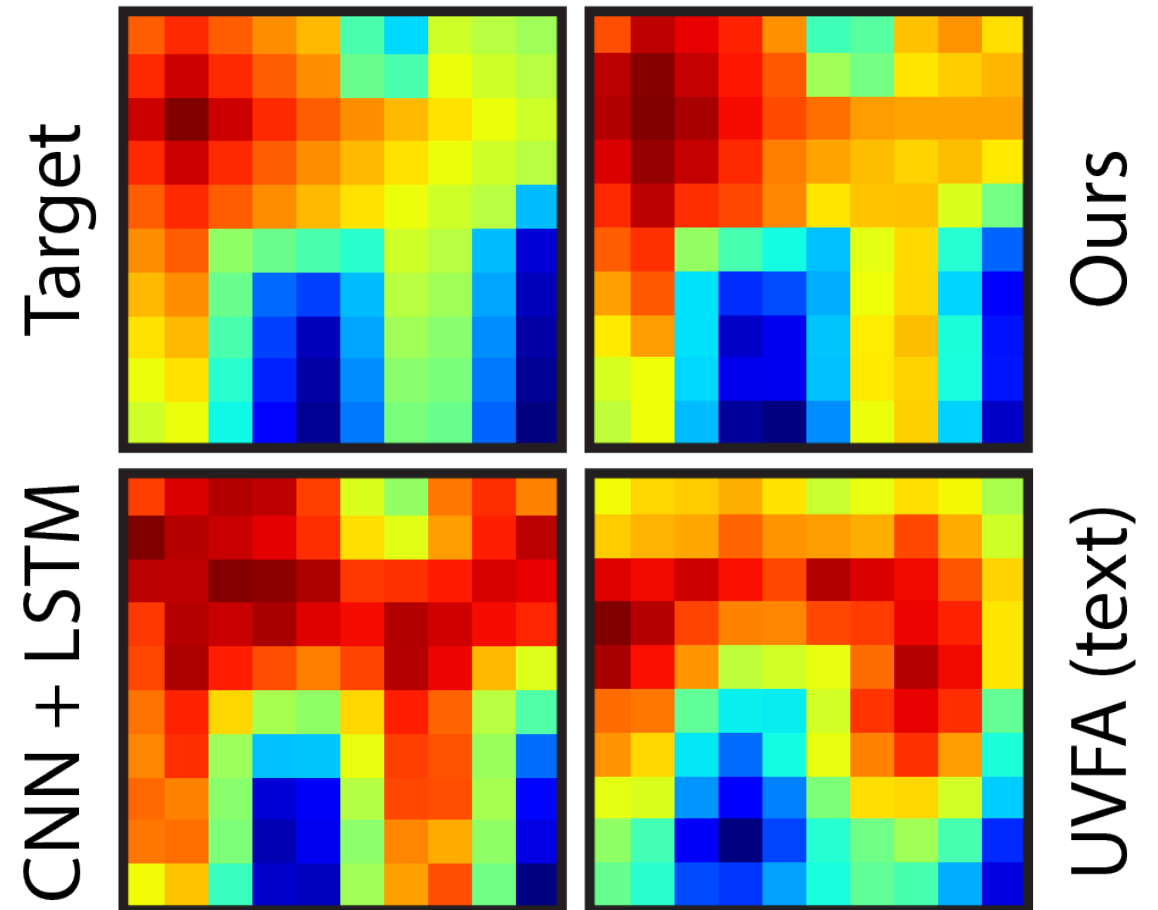
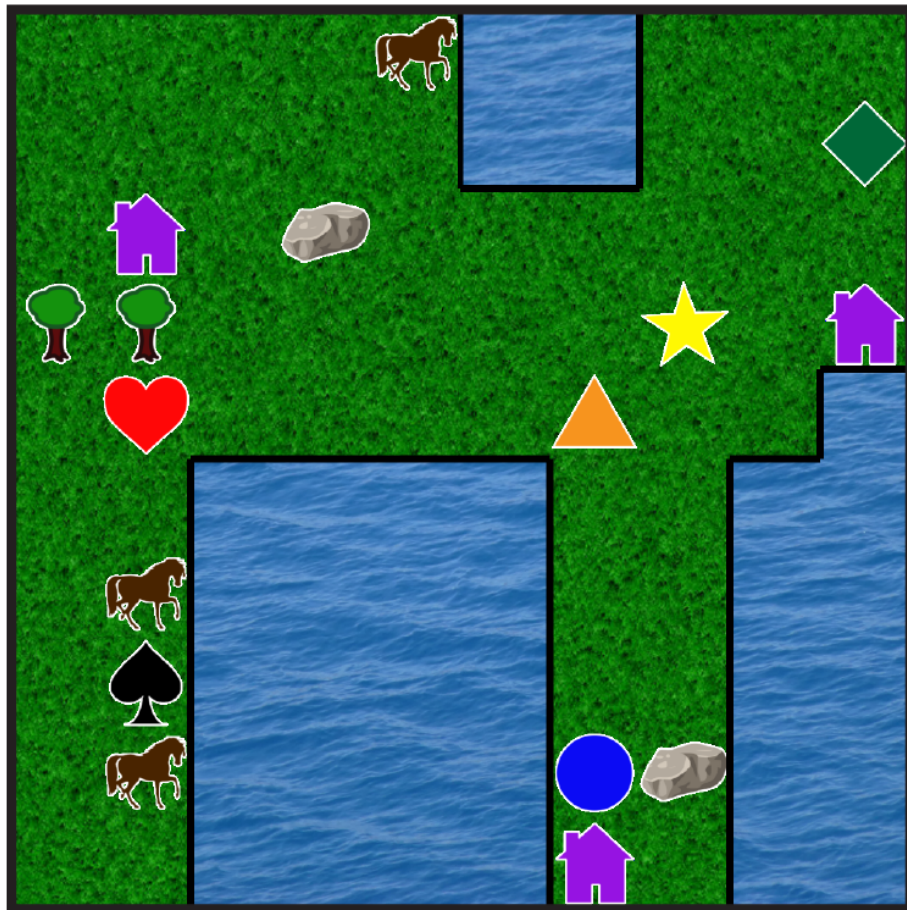


# Sample efficiency

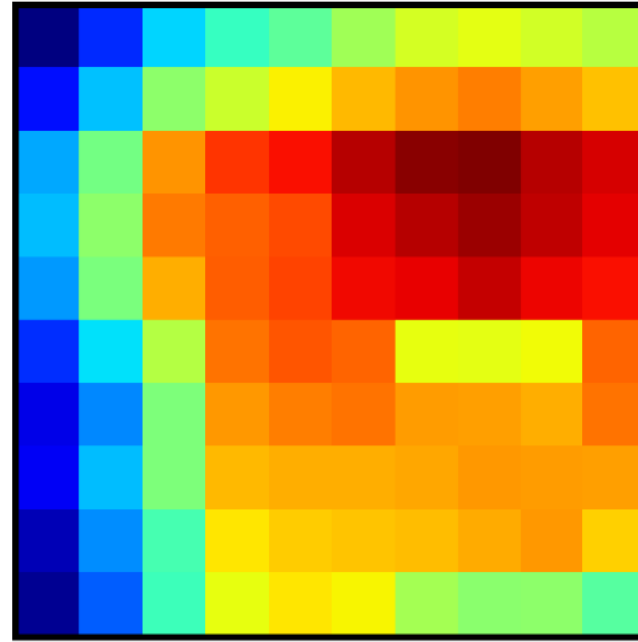
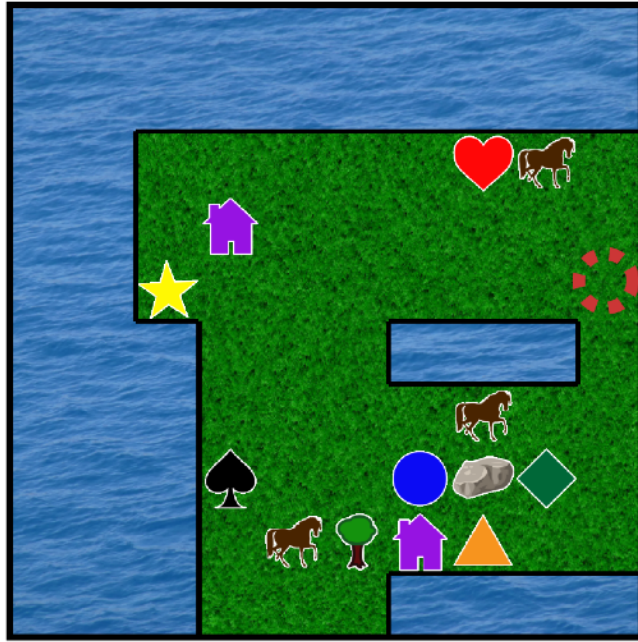


# Value maps

*Reach the northernmost house.*

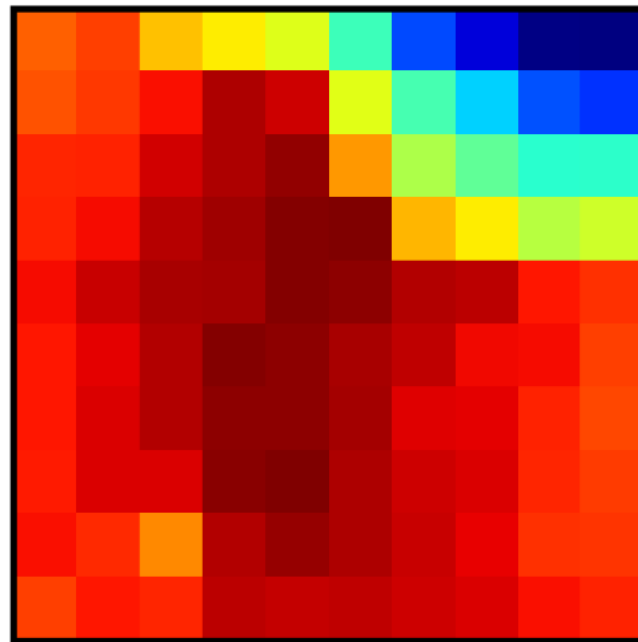
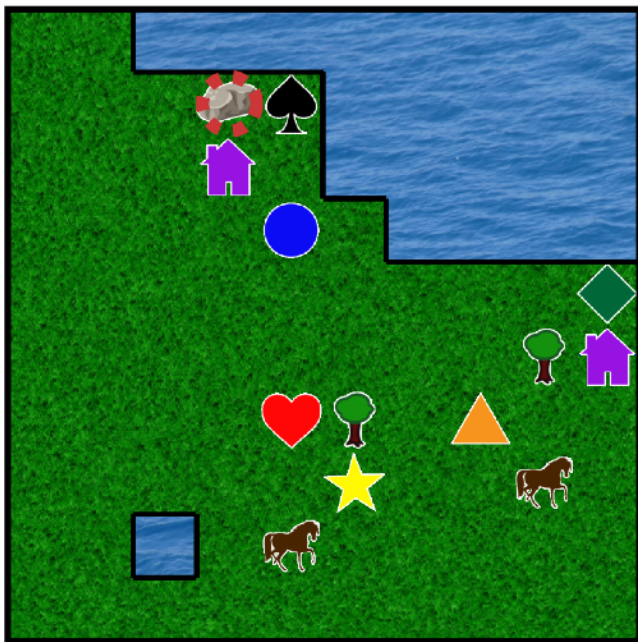






Multiple levels of indirection

*reach cell one to the right and two down from horse located to the right of the heart*



Redundant information

*locate the cell that is filled w/a rock to the left of the teal spade and above the purple house which is above and to the left of the blue circle*

# Conclusions

- Language provides a compact medium for encoding knowledge for policy transfer
- Model-aware methods more suitable for leveraging language for transfer.
- Spatial reasoning is highly contextual and a challenging grounding task.
- Joint reasoning over text and environment is crucial for effective generalization over unseen worlds and linguistic variety.

