Continual Learning for Grounded Language Generation by Observing Human Following Behavior

Noriyuki Kojima, Alane Suhr, and Yoav Artzi

EMNLP 2021 (TACL paper)

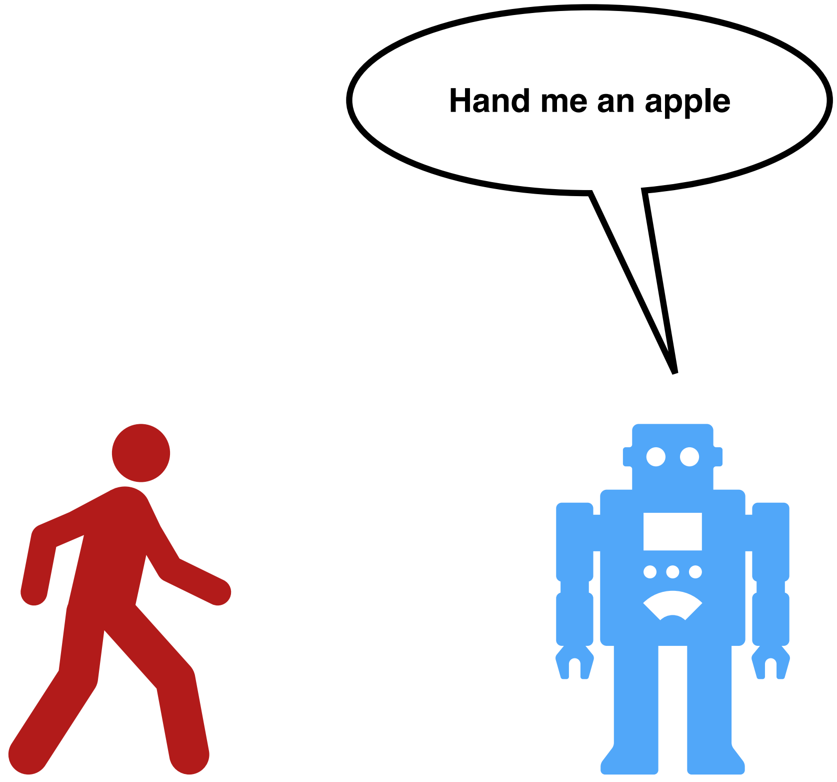
Cornell Bowers C·IS
Computer Science
CORNELL TECH

Task

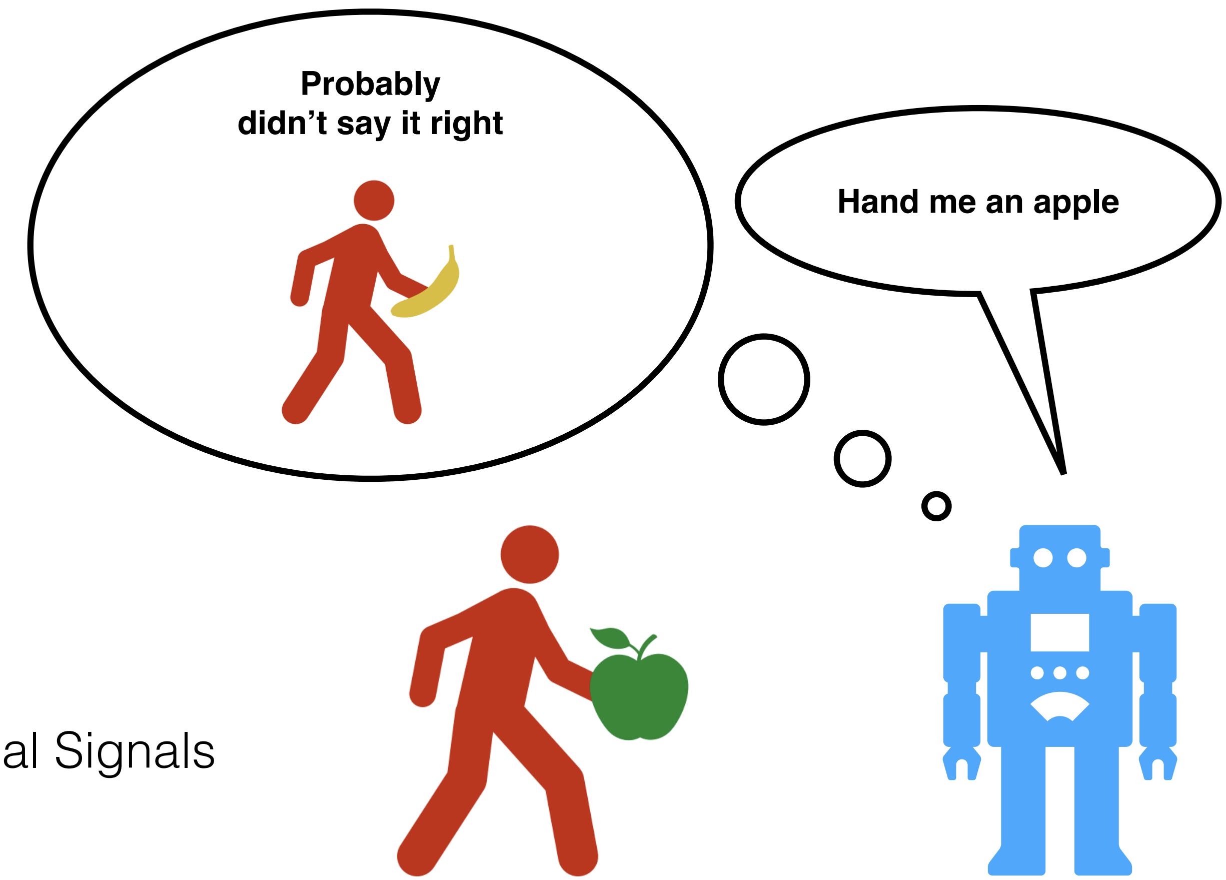
Learning a grounded instruction generation system

$$f(\text{world state, system intent}) = \text{instruction}$$

Learning Instruction Generation From Human Behavior

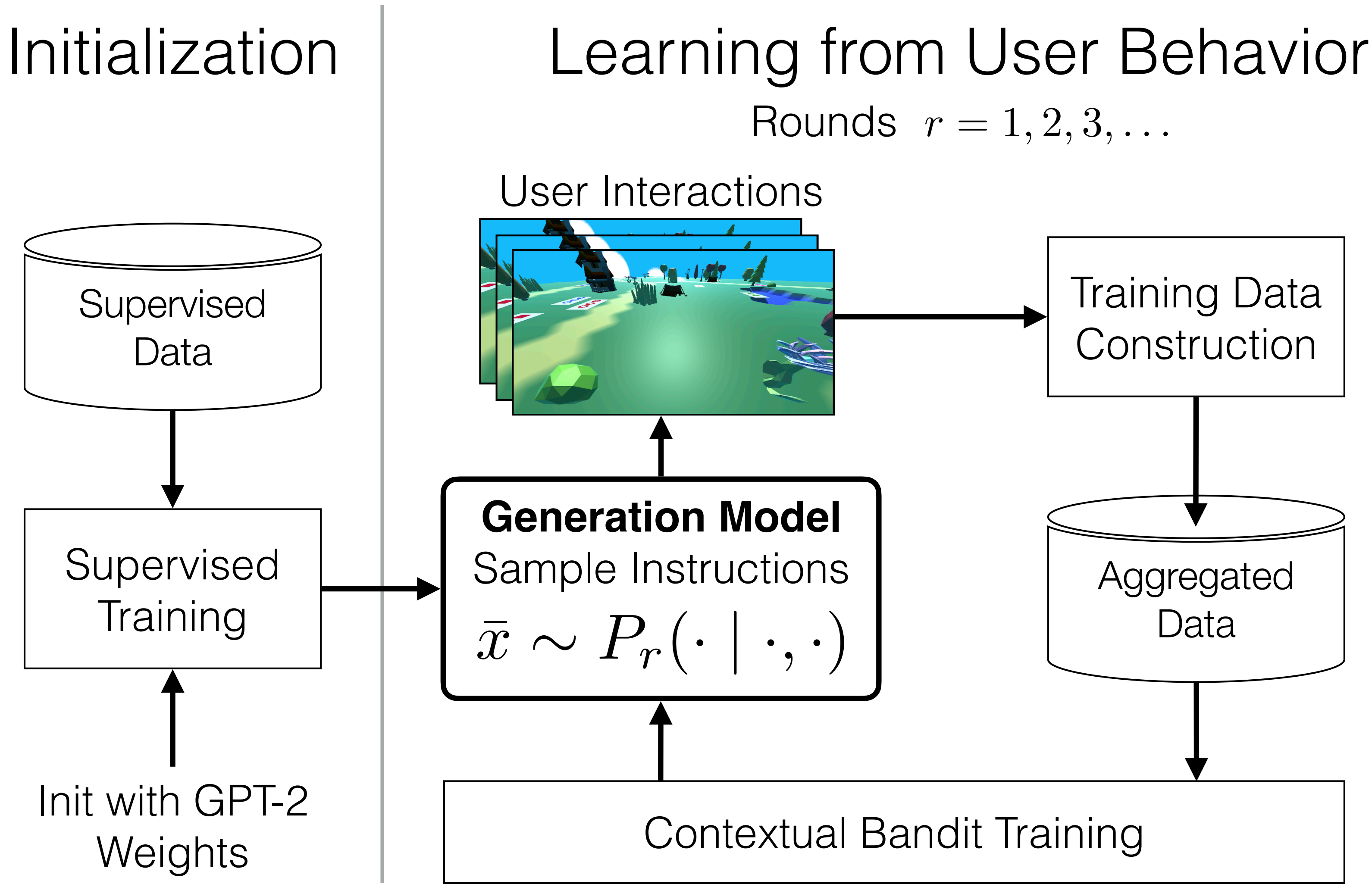


Learning Instruction Generation From Human Behavior



Observational Signals

Learning Overview

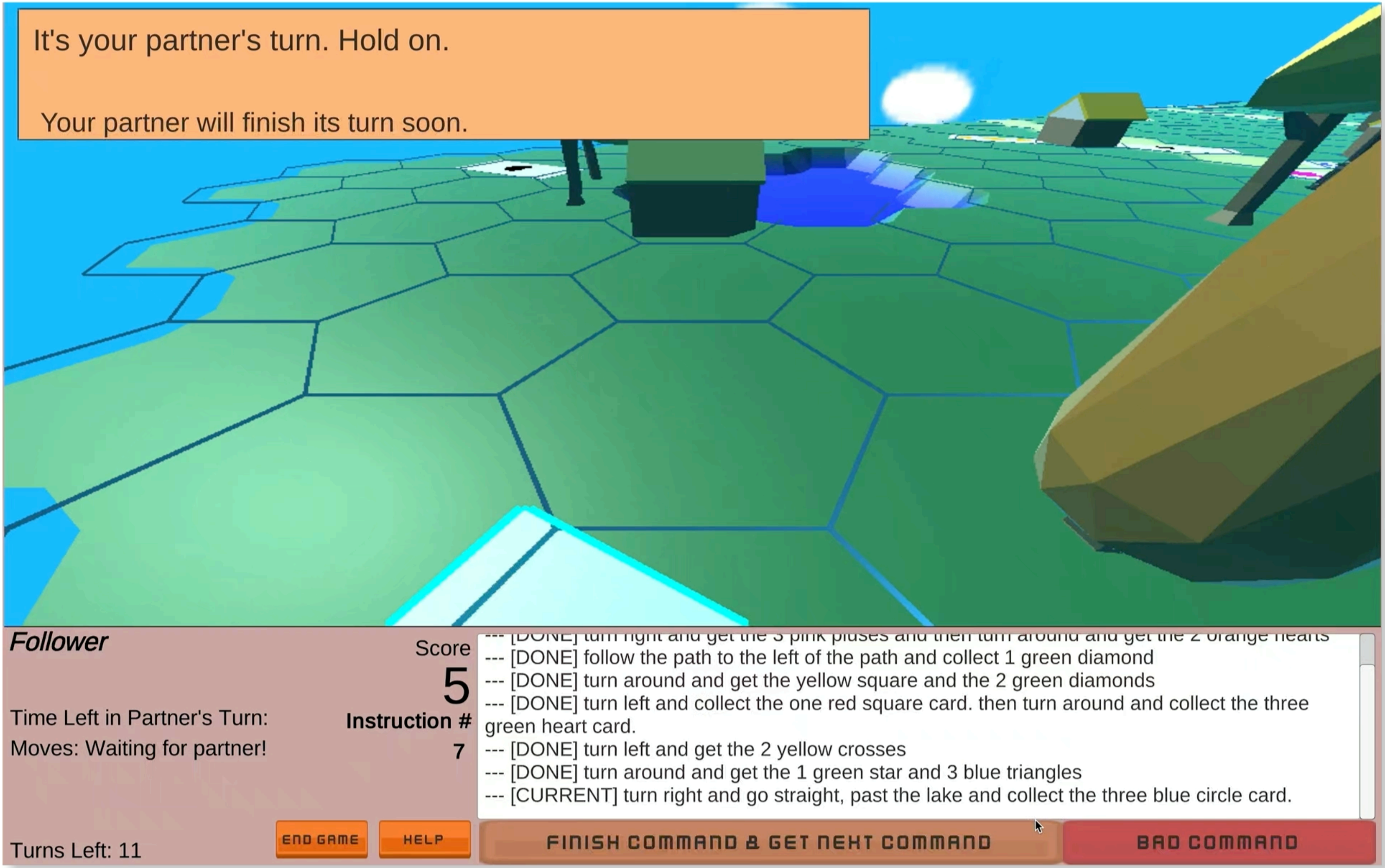


Continual Generation Learning in CerealBar

CerealBar is a situated collaborative game with sequential natural language instruction

- Two agents collaborating in an environment
- Goal: collecting card sets together
- Uni-directional natural language instruction
- System as a leader, human user as a follower

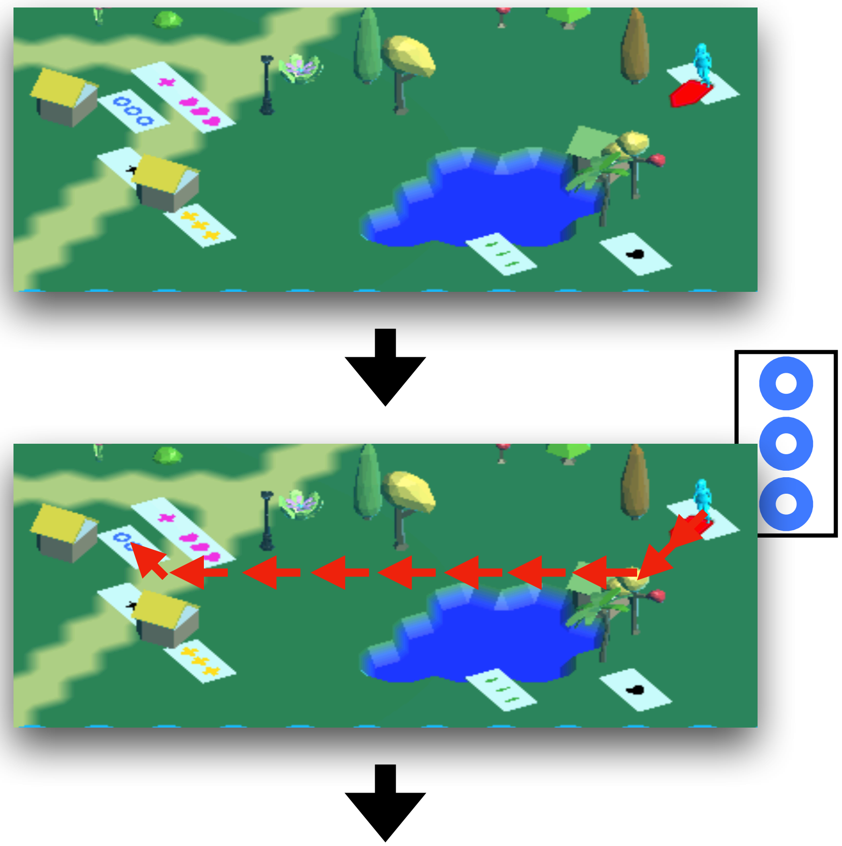
[Suhr et al., EMNLP 2019]



turn right and go straight, past the lake and collect the three blue circle card.

Generating Instructions in CerealBar

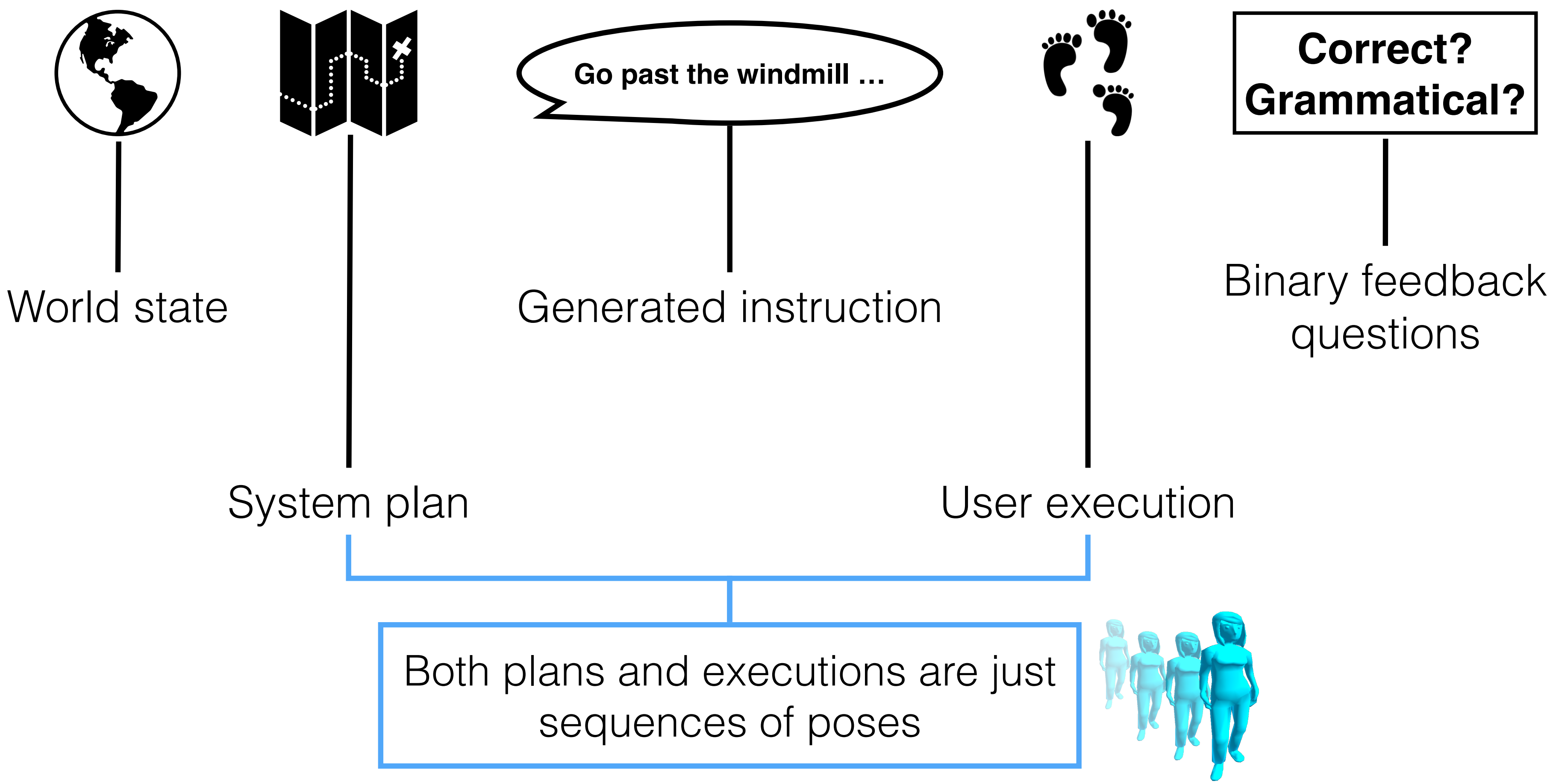
- **Input:** game state
- **Output:** instruction describing the follower's moves and target cards
- Which cards to select? → deterministic planner



Turn right and go straight, past the lake and collect the three blue circle card.

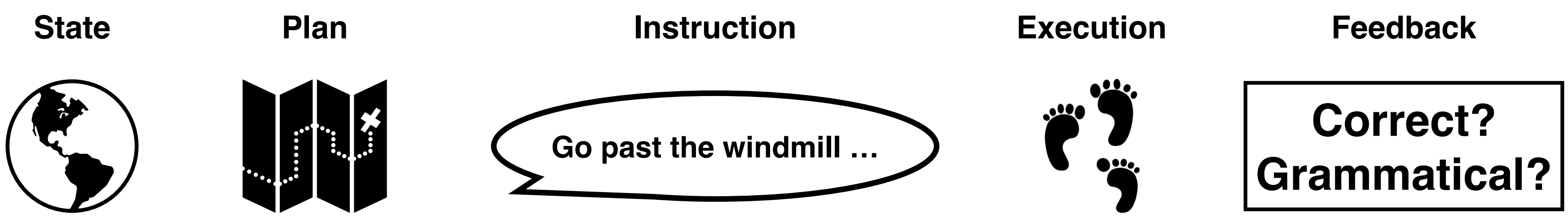
Interaction Data

For each user execution of a generated instruction:



Reward Computation

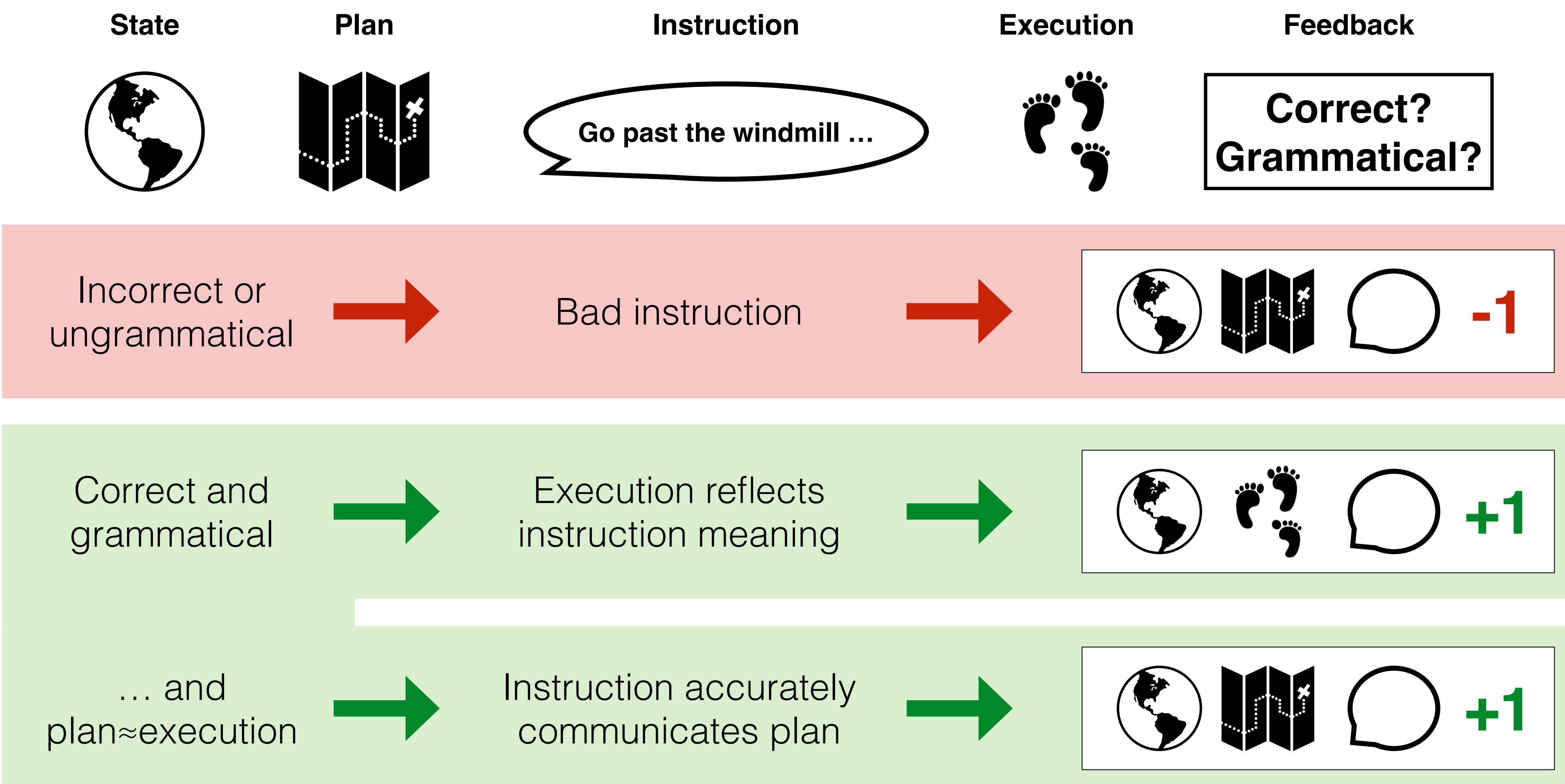
For each user execution of a generated instruction:



- Compare the system's plan to the user execution
- If they diverge, the instruction is not a good representation of the plan
- But, could still be a good representation of user execution

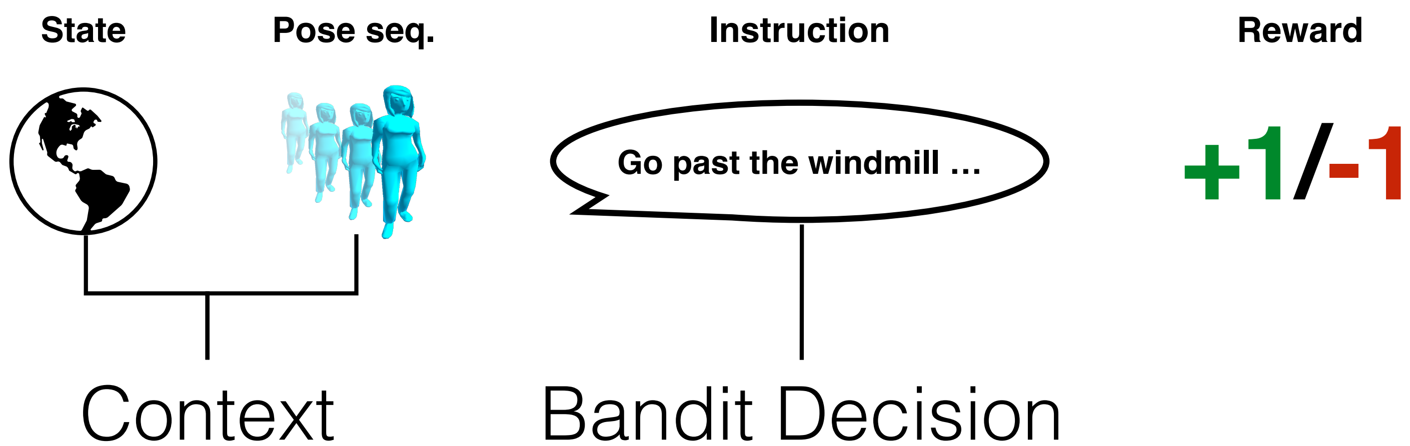
Reward Computation

For each user execution of a generated instruction:



Training Data

Each training example includes:



- A contextual bandit scenario
- State and pose sequence are contexts to generate the instruction, which gets a reward

Training Objective

State s	**Pose seq.** $\bar{\rho}$	**Instruction** $\bar{x}$	**Reward** y
			**+1/-1**

- Objective: maximize the reward
- Gradient is:

$$\nabla \mathcal{L} = \overset{\text{Reward/label}}{y} \nabla P(\overset{\text{Instruction}}{\bar{x}} \mid \underset{\text{State}}{s}, \underset{\text{Pose seq.}}{\bar{\rho}})$$

- Positive examples behave exactly like supervised learning
- Negative examples? $\lim_{P(\cdot) \to 0} \log P(\cdot) = -\infty$

Training Objective

State s **Pose seq.** $\bar{\rho}$ **Instruction** $\bar{x}$ **Reward** y

+1/-1

- Objective: maximize the reward + IPS for negative examples

- Gradient is:

Reward/label

Instruction

$$\nabla \mathcal{L} = \ell(y) y \nabla P(\bar{x} \mid s, \bar{\rho})$$

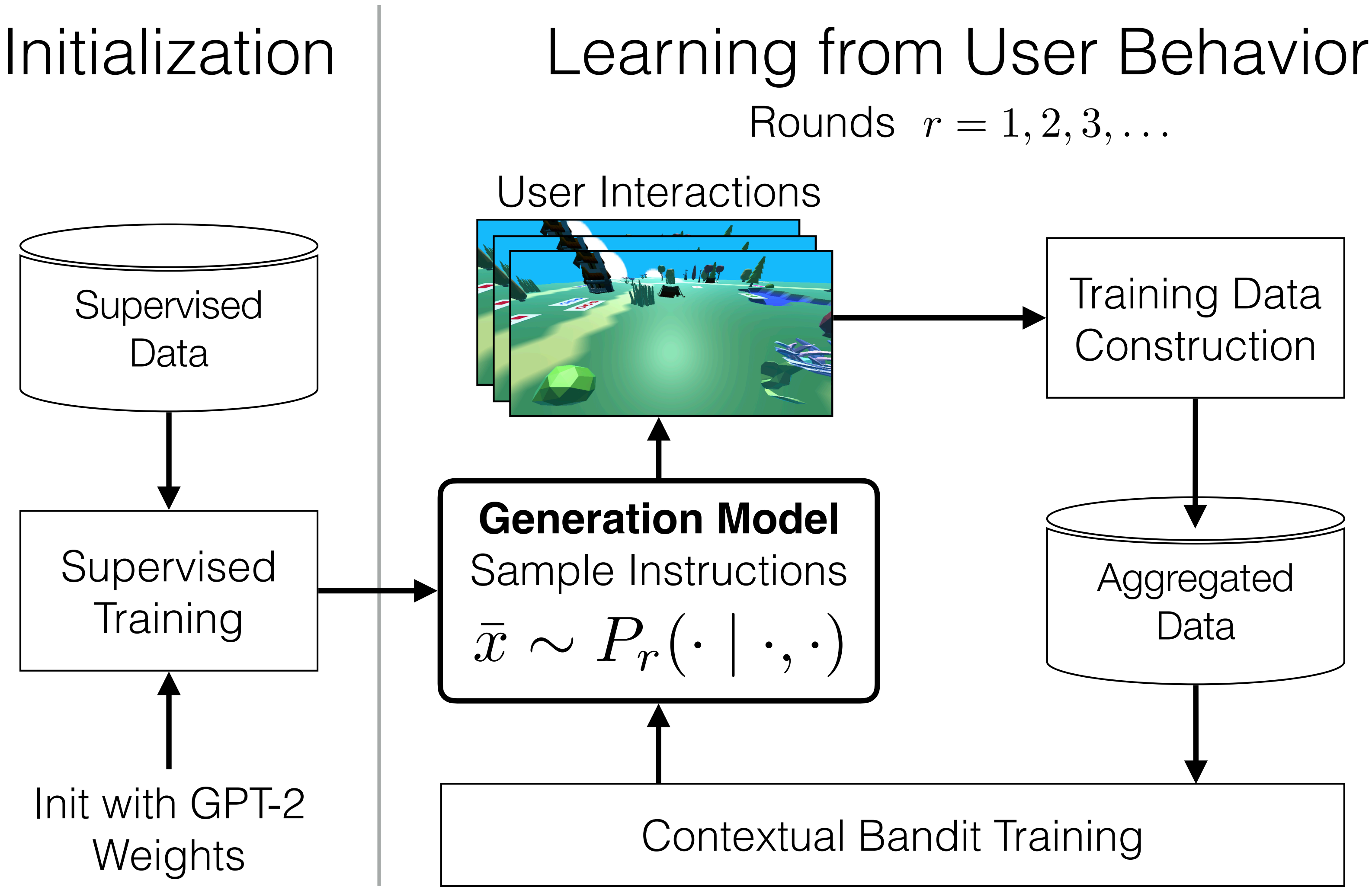
Pose seq.

State

$$\ell(y) = \begin{cases} 1 & y = +1 \\ \frac{P(\bar{x}|s,\bar{\rho})}{P'(\bar{x}|s,\bar{\rho})} & y = -1 \end{cases}$$

Original sampling probability

Putting it All Together



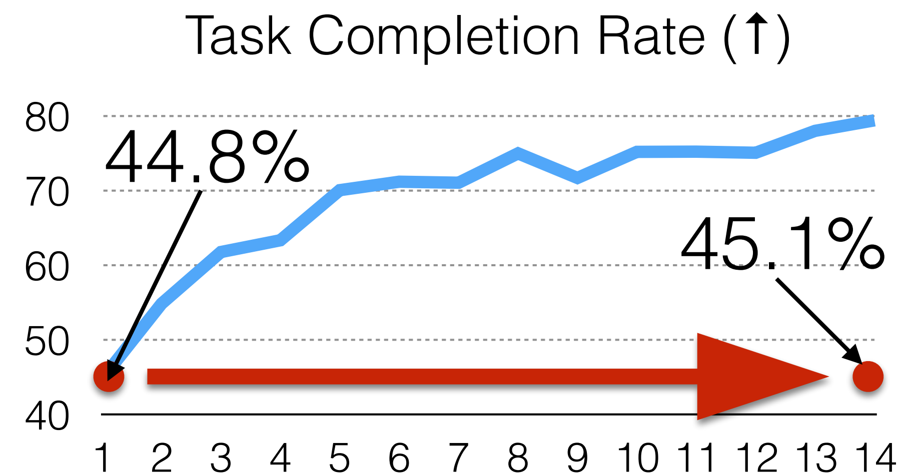
Model

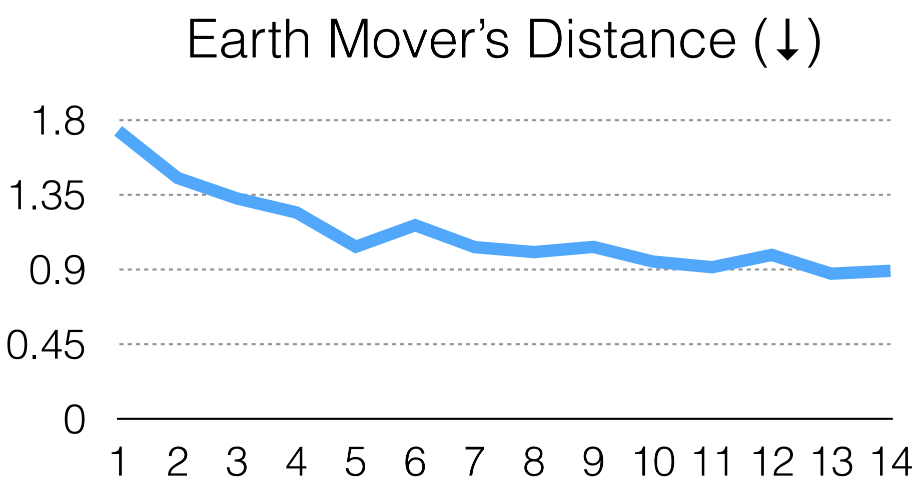
- Encoder-decoder architecture
- Spatial encoding of the environment and the system's plan (or execution) to a sequence of vectors
- GPT-2 Transformer decoder conditioned on encoder output via pseudo-self attention [Ziegler et al. 2019]

Experimental Setup

- Initialize the model using wizard-of-oz interactions
- Evaluate via user task completion and similarity of user execution to system’s plan using earth’s mover distance
- No good stopping criteria, so just train for fixed number of epochs

Long-term Study



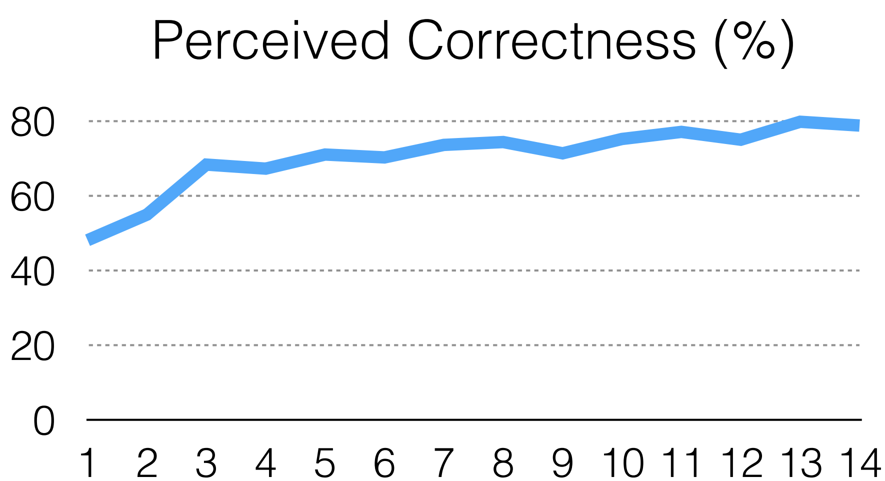


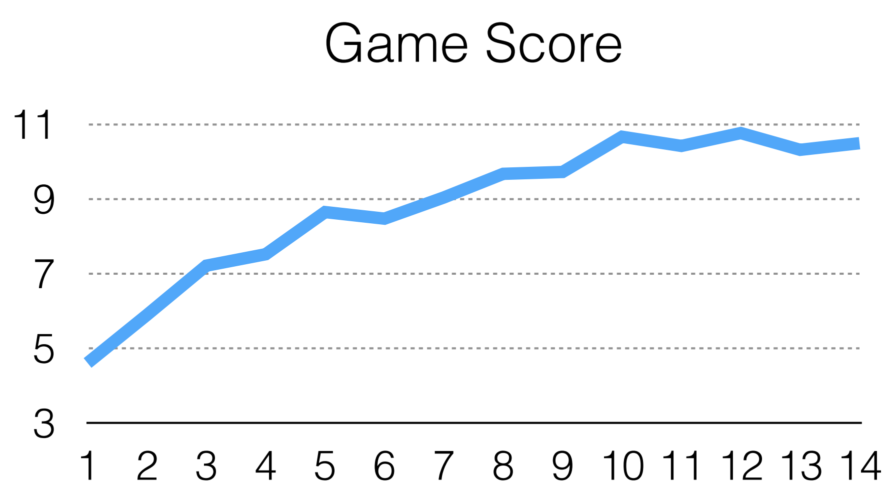
- The model continually improves in generating instructions that relay its intent

- Task completion improves 44.8→79.4%

- User adaptation does not contribute to system's improvement

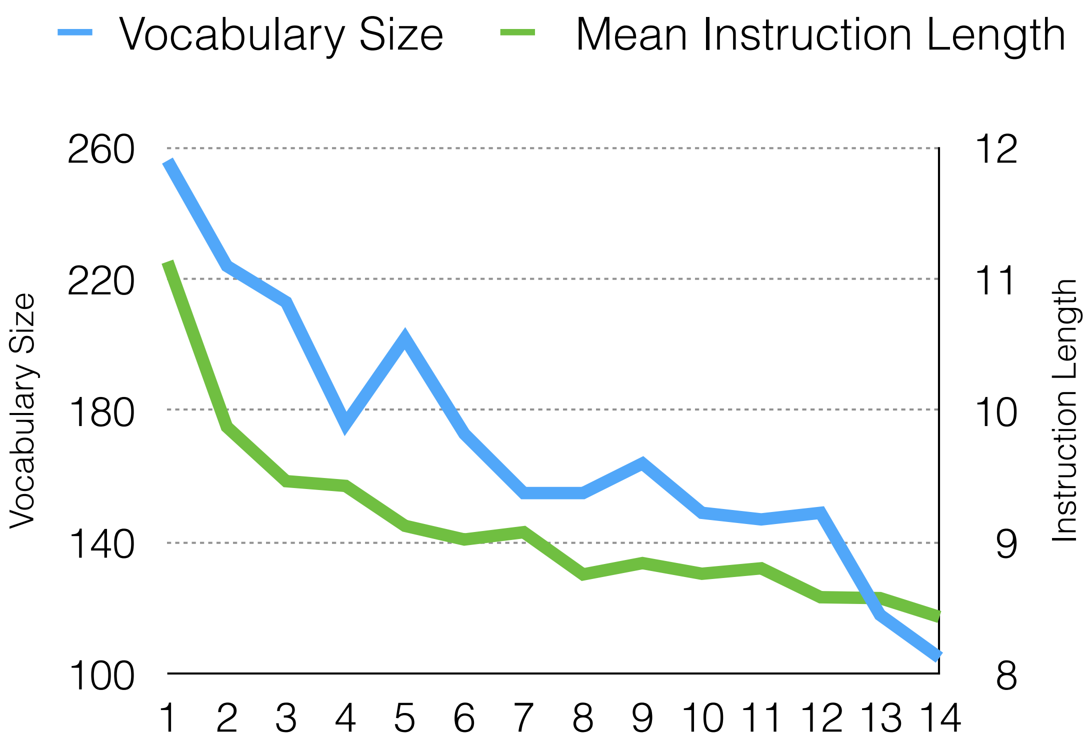
Long-term Study





- Users' perception of the correctness of their actions with respect to system intent improves
- Overall system performance improves 4.5→10.4 points

Long-term Study



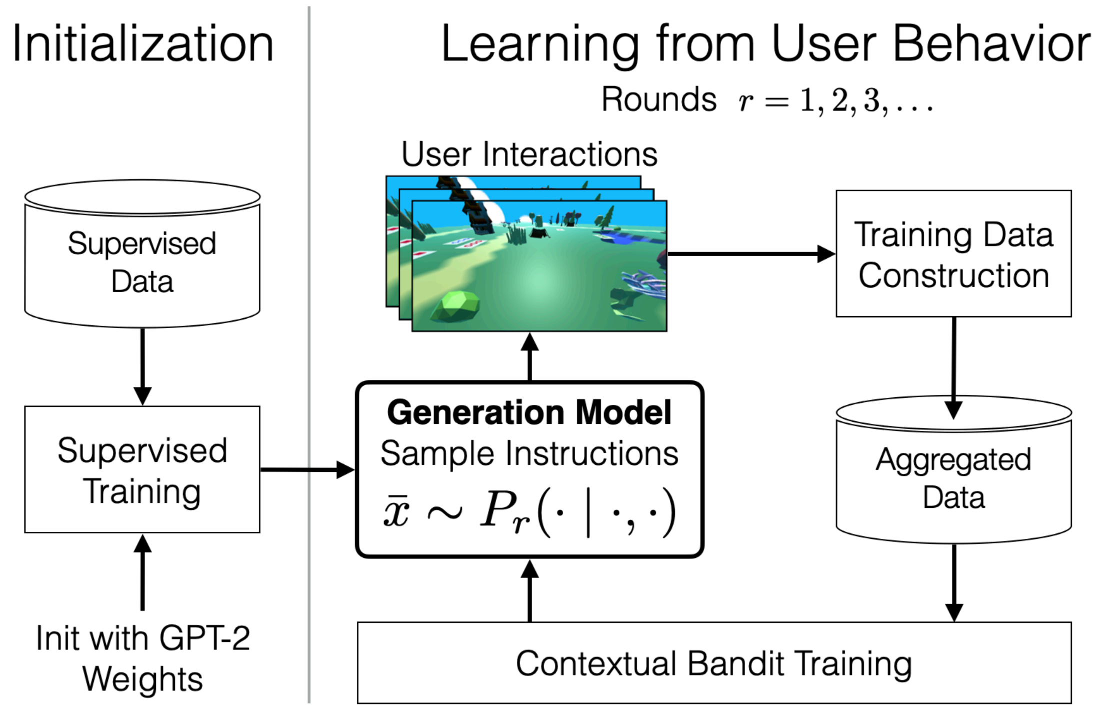
- Language becomes simpler
- Potentially more attuned to the task
- But some side-effects

Further Experimental Highlights

- Error analysis shows reduction of all error categories, such as specifying incorrect cards

- Study shows learning signal is robust across different learning designs

- More results: task-complexity breakdown results, comparison to supervised learning …



lil.nlp.cornell.edu/cerealbar

Noriyuki Kojima

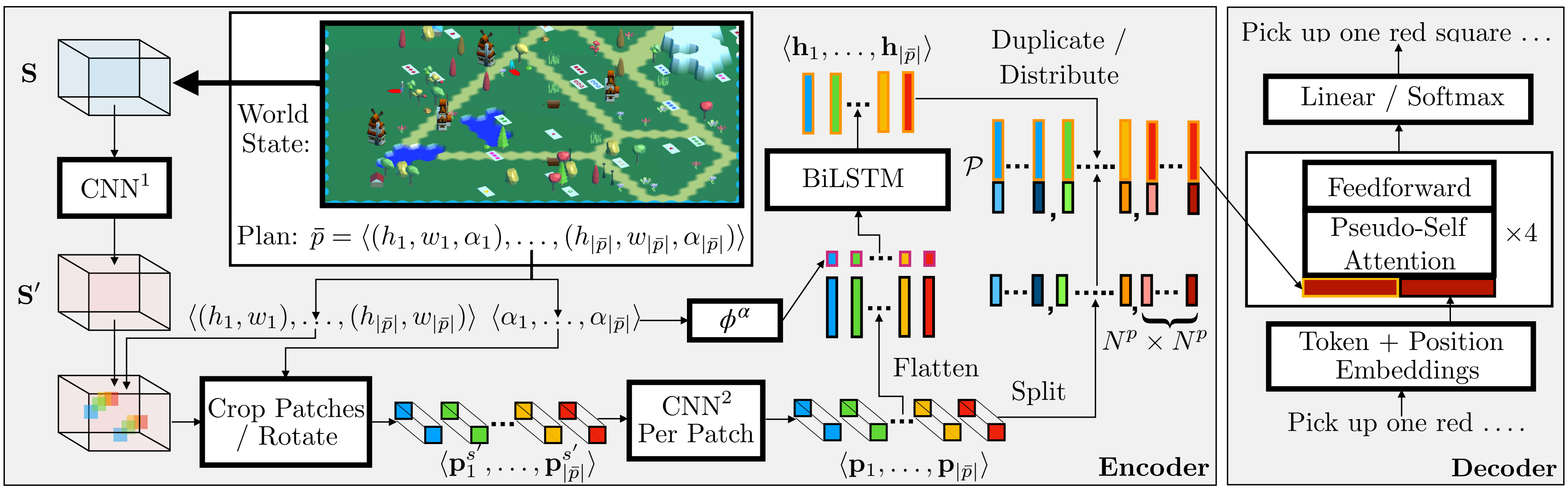
Alane Suhr

Yoav Artzi

Thank you! Questions?

Supplementary Slides

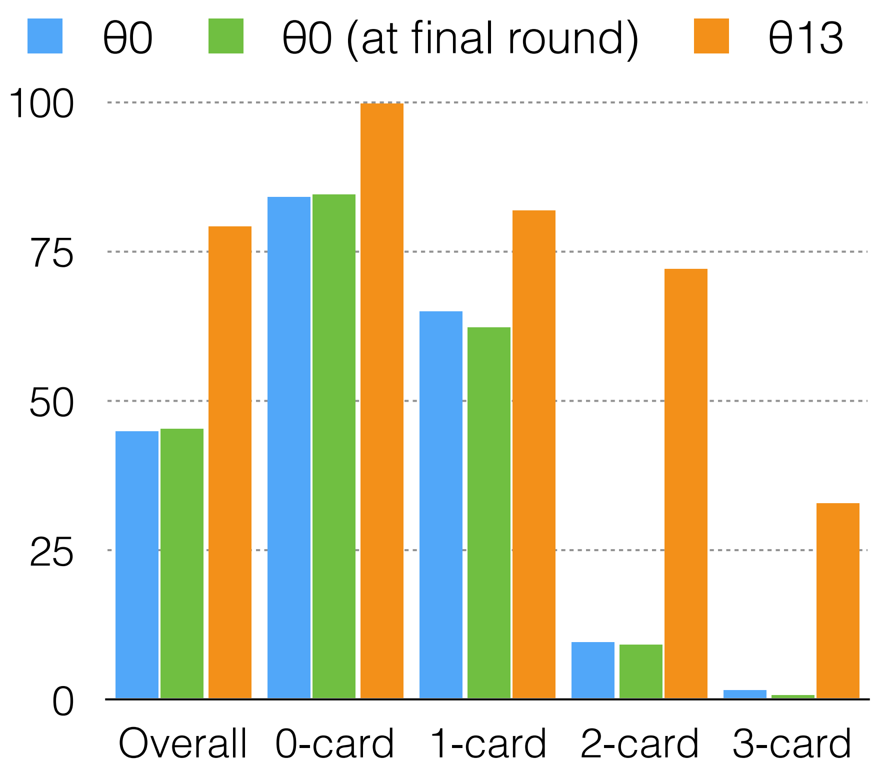
Model

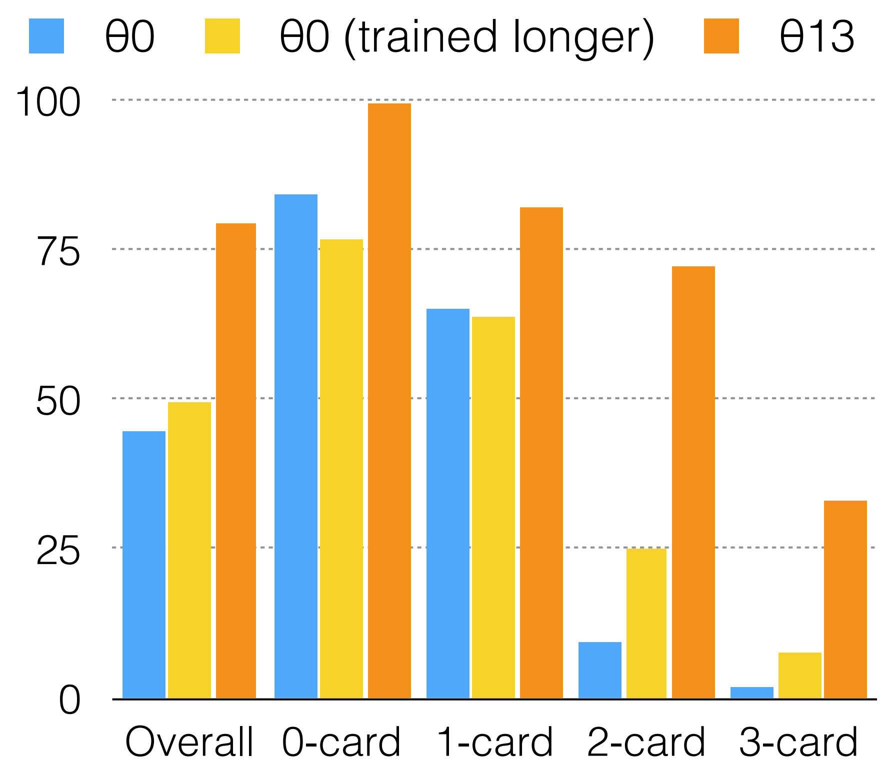


S
S′
CNN^1
World State:
Plan: $\bar{p} = \langle (h_1, w_1, \alpha_1), \ldots, (h_{|\bar{p}|}, w_{|\bar{p}|}, \alpha_{|\bar{p}|}) \rangle$
$\langle (h_1, w_1), \ldots, (h_{|\bar{p}|}, w_{|\bar{p}|}) \rangle$
$\langle \alpha_1, \ldots, \alpha_{|\bar{p}|} \rangle$
ϕ^α
Crop Patches / Rotate
$\langle \mathbf{p}_1^{s'}, \ldots, \mathbf{p}_{|\bar{p}|}^{s'} \rangle$
CNN^2 Per Patch
$\langle \mathbf{p}_1, \ldots, \mathbf{p}_{|\bar{p}|} \rangle$
Flatten
BiLSTM
$\langle \mathbf{h}_1, \ldots, \mathbf{h}_{|\bar{p}|} \rangle$
Duplicate / Distribute
$\mathcal{P}$
Split
$N^p \times N^p$
Encoder
Pick up one red square . . .
Linear / Softmax
Feedforward
Pseudo-Self Attention
×4
Token + Position Embeddings
Pick up one red . . . .
Decoder

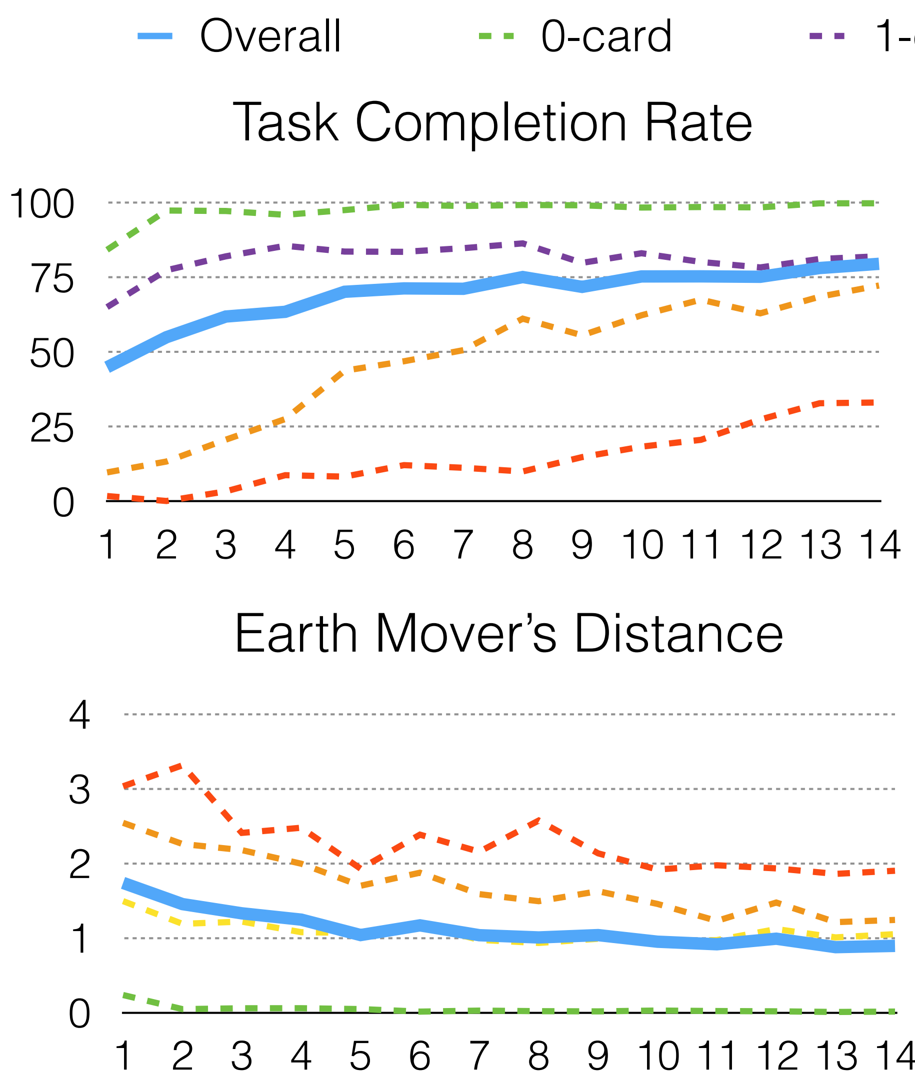
Confounding Factors?

- User Adaptation?
- Training longer (i.e., training stopping criteria)?



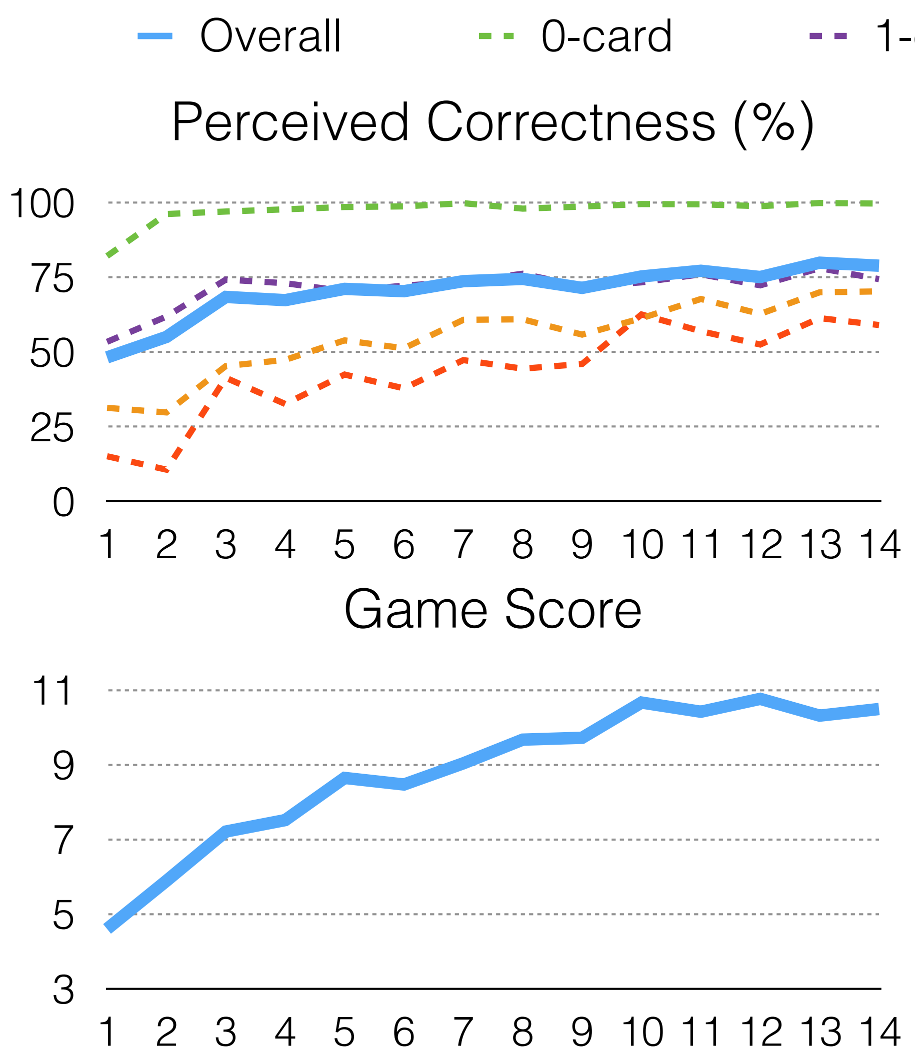


Long-term Study: 14 Rounds



- The model continually improves in generating instructions that relay its intent
- Task completion improves 44.7→79.3%
- Mulit-goal instructions take longer to improve, but accelerate later on

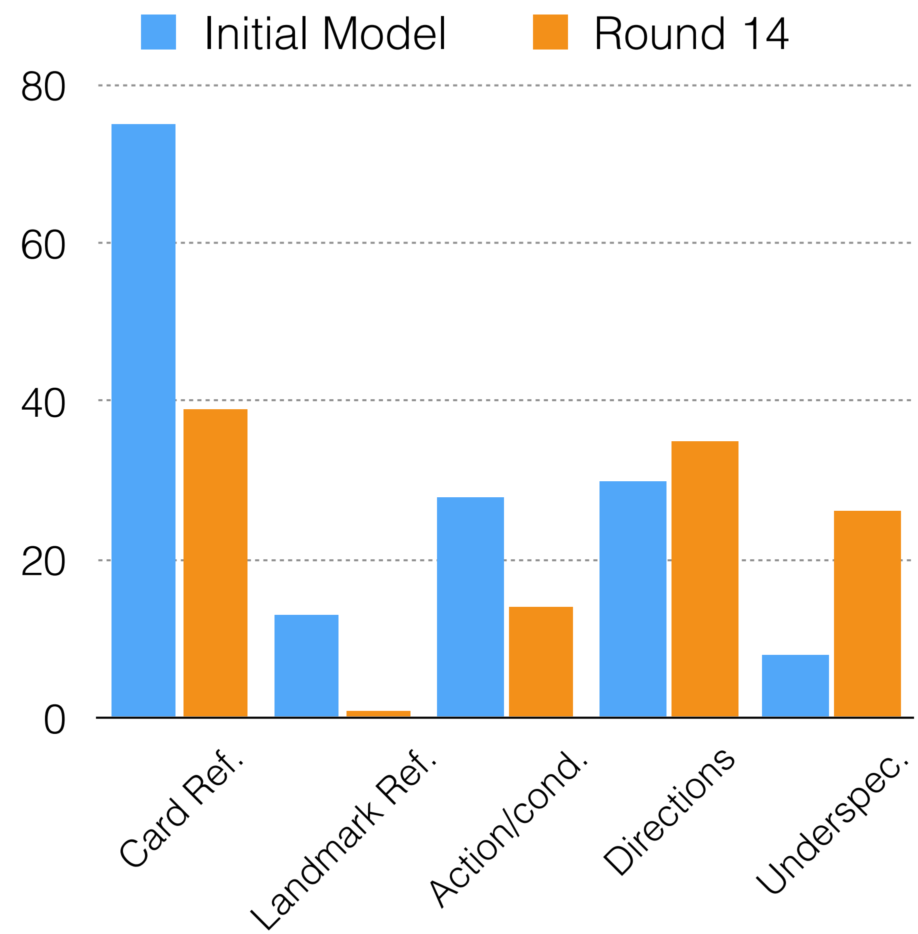
Long-term Study: 14 Rounds



- Users' perception of the correctness of their actions with respect to system intent improves
- Overall system performance improves (4.5→10.4 points)

Error Analysis

- Overall proportion of errors decreased 68.5→26.8%
- Manually analyzed 100 erroneous instructions from initial and final rounds
- Improvements across all error categories
- Share of errors that are underspecifications increases, potentially because of the smaller vocabulary



Error Analysis

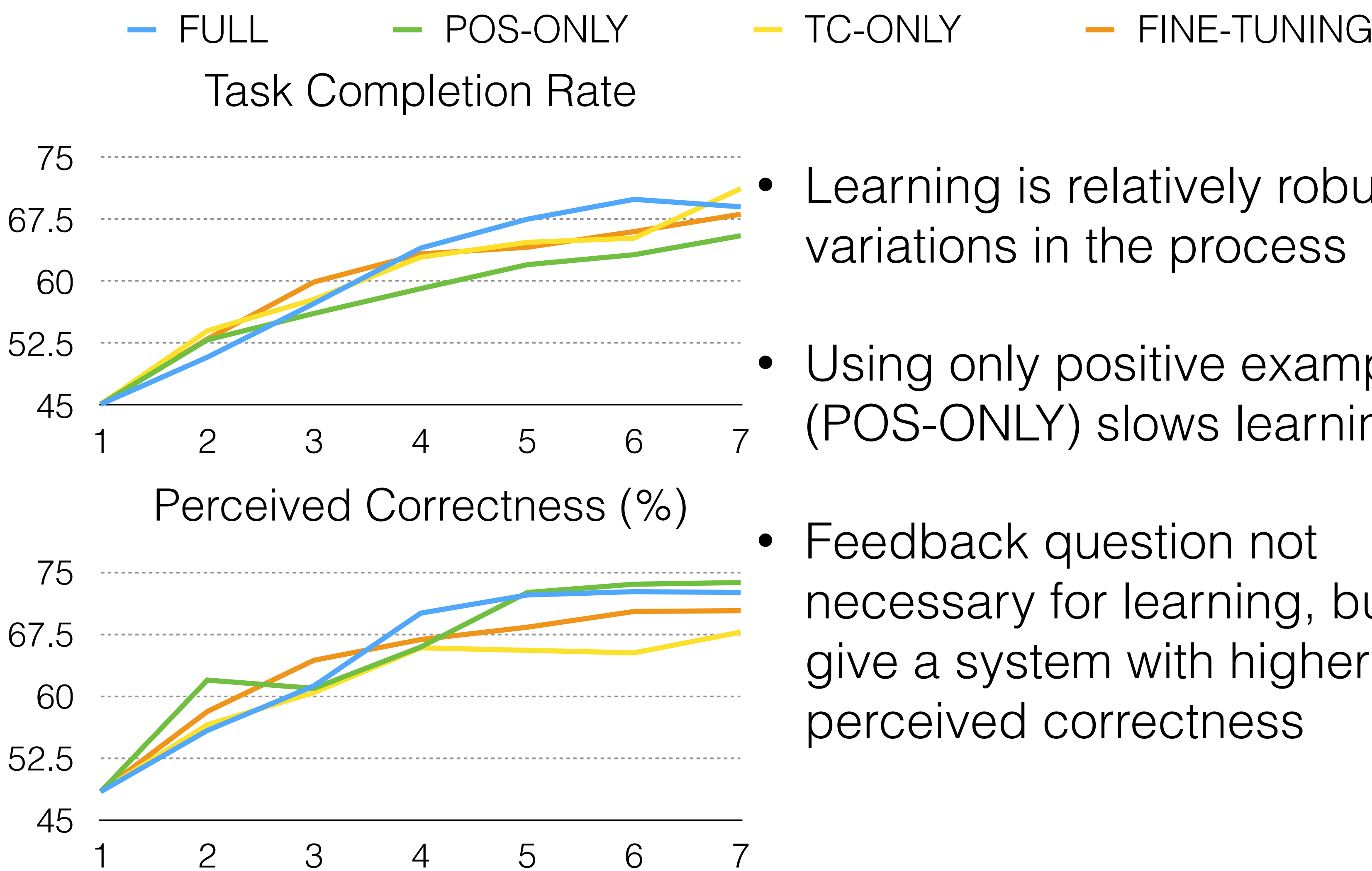
Error Type	$r = 1$	$r = 14$	Example
Incorrect, missing, or extra cards	75	39	*turn left and go to the yellow ~~star~~ triangles*
Irrelevant landmarks	13	1	*Head toward the ~~windmill~~ house. grab 2 red and triangle*
Incorrect direction	30	35	*grab the black heart to ~~your left~~ in front of you.*
Incorrect actions or conditions	28	14	*~~After the two red triangles~~, get the 3 red triangles.*
Underspecification	8	26	*turn right and go straight toward red trees collect two orange triangle.*
Implausible instructions	11	1	*Turn left and get the two pink hearts ~~and the two pink hearts near the pink hearts.~~*
Proportion of erroneous instructions	68.5%	26.8%	

Table 1: The types of errors observed in erroneous instructions generated during the first ($r = 1$) and final ($r = 14$) rounds of deployment. We show error counts from the 100 randomly-sampled erroneous instructions. Examples illustrate error categories; red strikethrough shows erroneous segments, and blue fragments show possible corrections. Instructions that fit into multiple categories are double counted.

System Variants Study

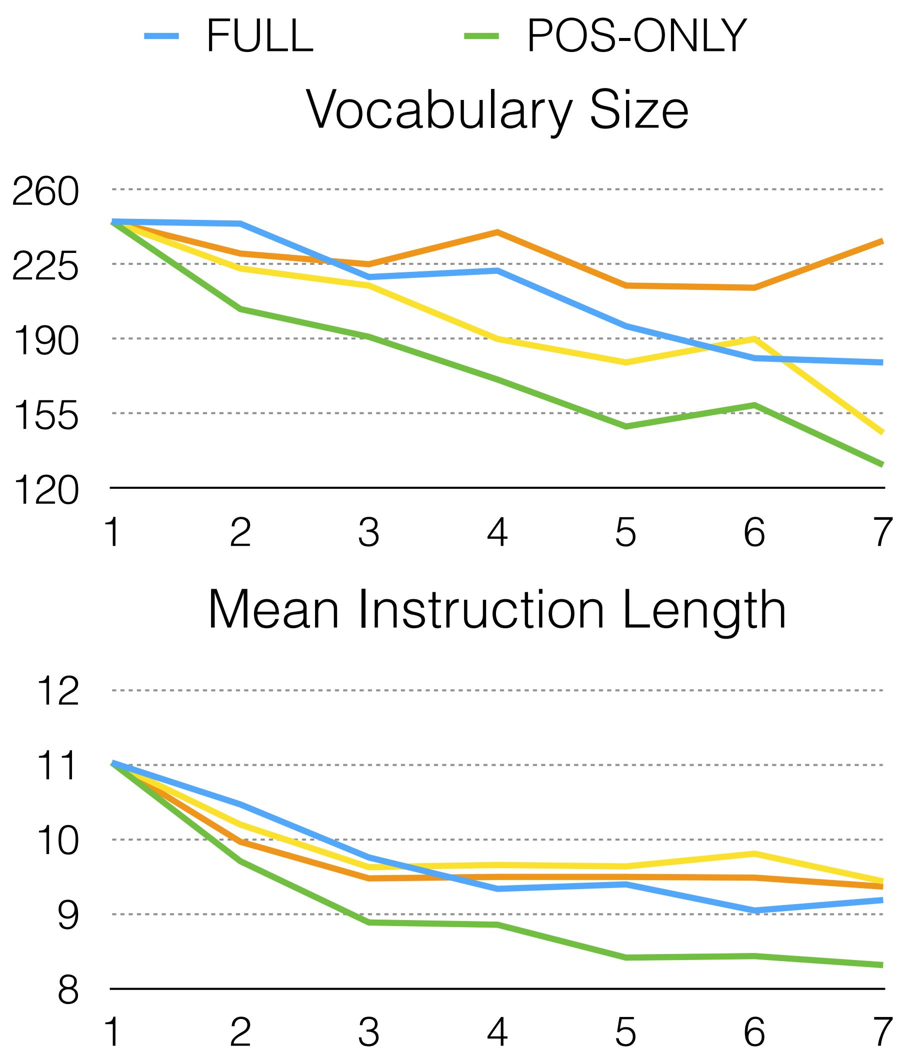
- **FULL:** basic setup
- **POS-ONLY:** use only examples with positive labels
- **TC-ONLY:** ignore feedback questions, assign positive labels if the user completes the task
- **FINE-TUNING:** fine-tune w/rehearsal instead of training from scratch

System Variants Study



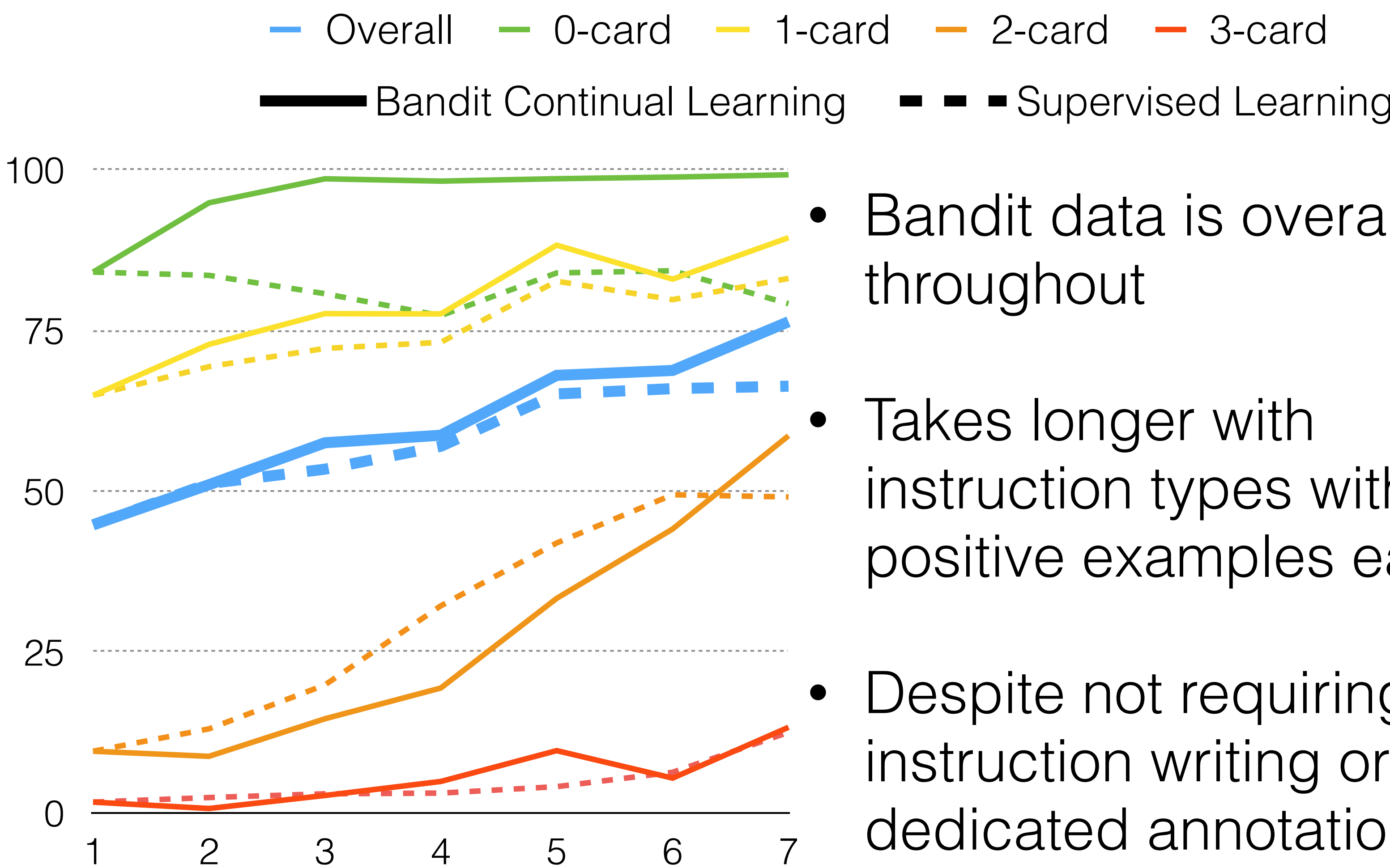
- Learning is relatively robust to variations in the process
- Using only positive examples (POS-ONLY) slows learning
- Feedback question not necessary for learning, but give a system with higher perceived correctness

System Variants Study



- Fine-tuning may be better at maintaining a more diverse vocabulary
- Without negative examples (POS-ONLY) length and vocabulary reductions is faster

Comparison to Supervised Learning



- Bandit data is overall better throughout
- Takes longer with instruction types with few positive examples earlier on
- Despite not requiring any instruction writing or dedicated annotation