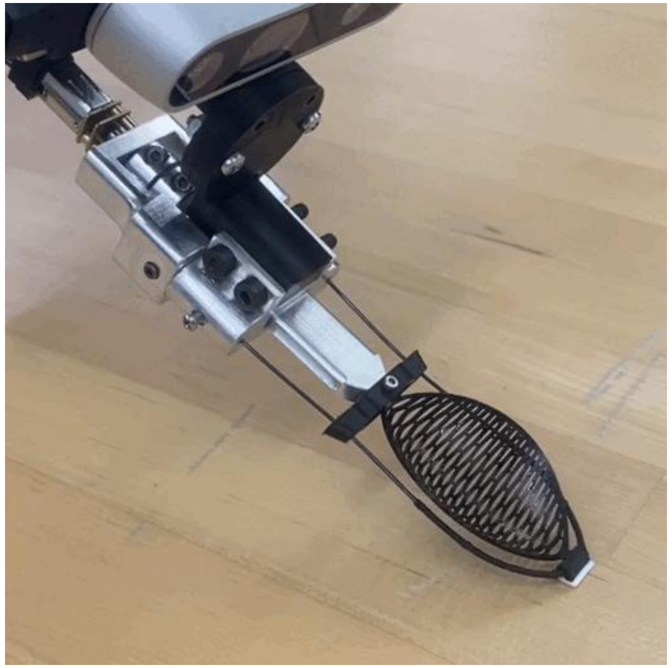
Simplifying Robot Personalization

Heramb Nemlekar

Assistant Professor

CSUN® | CALIFORNIA STATE UNIVERSITY NORTHRIDGE

Assistive robotics



Kiri-Spoon

Barriers to state-of-the-art robotics

Rigaud et al. (2024) Journal of Rehabilitation and Assistive Technologies Engineering

Useful

Reliably perform variety of tasks

Adaptive

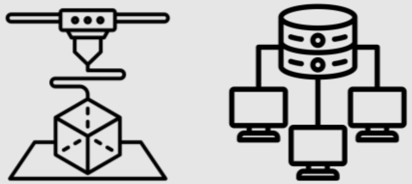
Personalize to individual needs

Easy

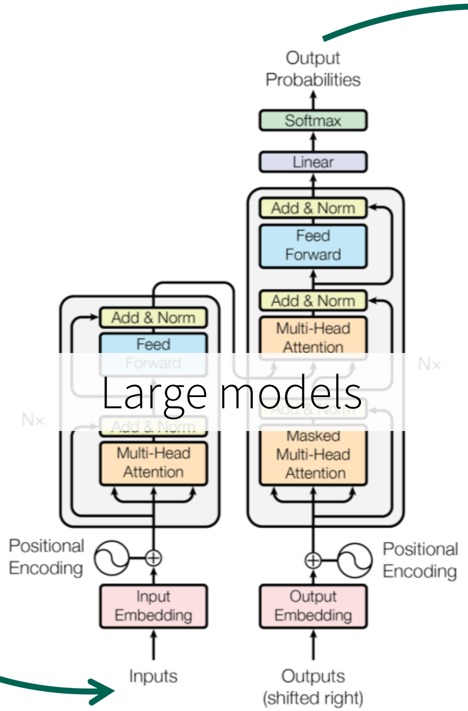
Intuitive with minimal training

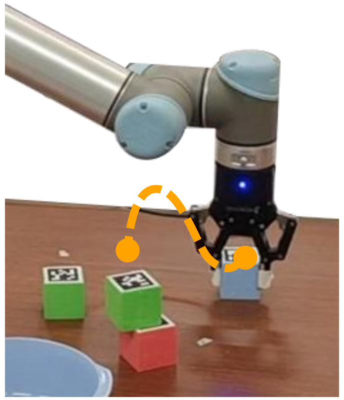
Accessible

Affordable hardware & compute



Imitation learning

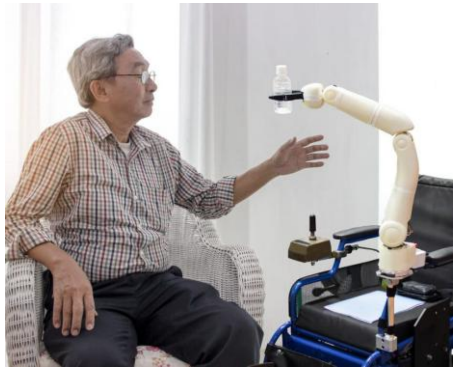


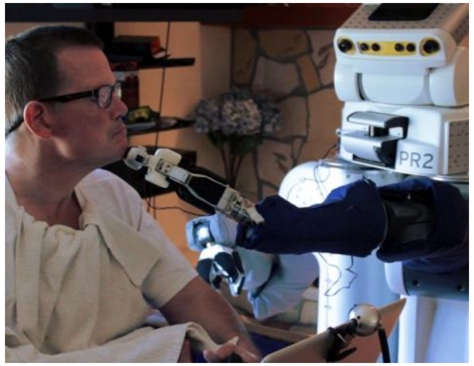


Imitate human demonstrations

Personalization

Unique individual needs of humans.





Many demonstrations? Different objects? Different heights?

Gap between end-users and state-of-the-art robot learners.

Barriers

Rigaud et al. (2024) Journal of Rehabilitation and Assistive Technologies Engineering

Useful

Reliably perform variety of tasks

Adaptive

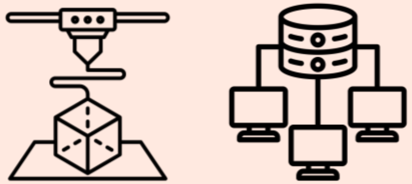
Personalize to individual needs

Easy

Intuitive with minimal training

Accessible

Affordable hardware & compute



Simplifying robot learning

How can we make it ***easy*** for humans ***to program robots***?

Forms of Input

Verbal EEG

Video recording

data

Finetuning

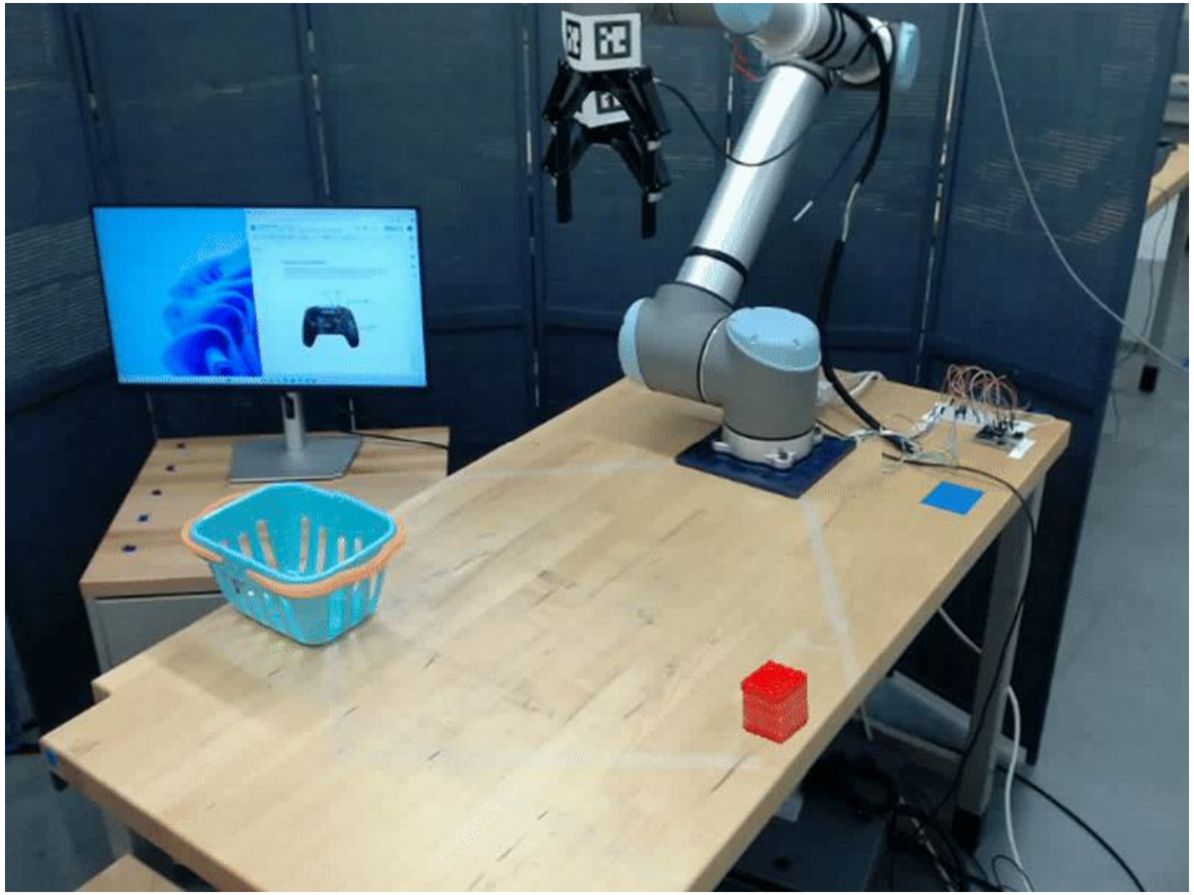
Dominant objectives

Prior

Human-human data

Teaching by drawing

L2D2: Robot Learning from 2D Drawings (AuRo 2025)



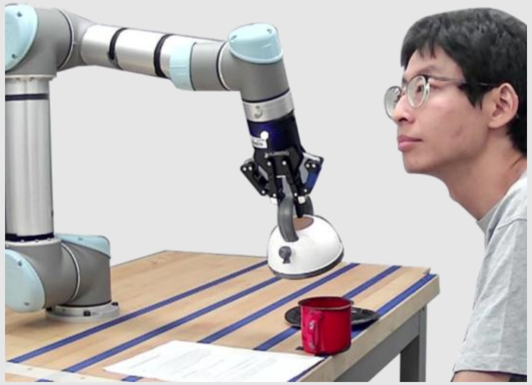
Simplifying robot learning

How can we make it ***easy*** for humans ***to program robots***?

Learning from
simpler tasks

Collaborative assembly

Using intuitive
tools & interfaces



Household tasks

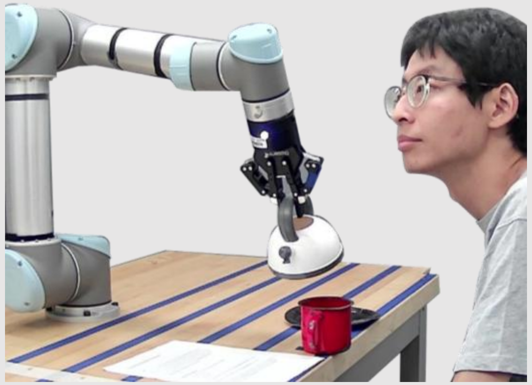
Simplifying robot learning

How can we make it ***easy*** for humans ***to program robots***?

Learning from
simpler tasks

Collaborative assembly

Using intuitive
tools & interfaces

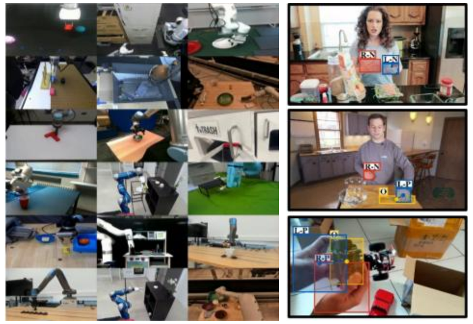


Household tasks

Imitation learning

Training on data from multiple tasks

common



small-scale, repetitive

VIOLA: Zhu et al. (2022); HYDRA: Belkhale et al. (2023); MimicPlay: Wang et al. (2023); Octo: Ghosh et al. (2024); RT2: Google DeepMind (2023); Track2Act: Bharadwaj et al. (2024); Open x-embodiment: O'Neill et al. (2024)

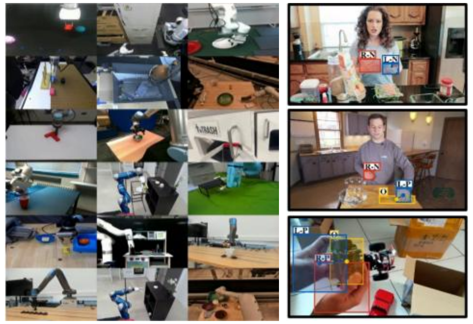
- Of course, to be able to perform these multiple tasks.
- Similarities and dissimilarities between the tasks can help the robot generalize beyond the training tasks.

Transfer *or extend knowledge* ***to new tasks****!*

Imitation learning

Training on data from multiple tasks

common



small-scale, repetitive

VIOLA: Zhu et al. (2022); HYDRA: Belkhale et al. (2023); MimicPlay: Wang et al. (2023); Octo: Ghosh et al. (2024); RT2: Google DeepMind (2023); Track2Act: Bharadwaj et al. (2024); Open x-embodiment: O'Neill et al. (2024)

What do you think is important for ***generalization***?

Data quality > More data

Research question

If you can choose one task to train the robot in, which one would you choose?

Towards transferring human preferences from canonical to actual tasks (RO-MAN 2022);
Transfer learning of human preferences for proactive assistance (HRI 2023 Best Paper finalist);
Selecting source tasks for transfer learning of human preferences (RA-L 2024)

Choosing training task

Say you want to teach a robot to play pickleball, which other training task will you choose?

Tennis (Similar S, A, R, T)
more difficult

Table tennis
Similar & easier

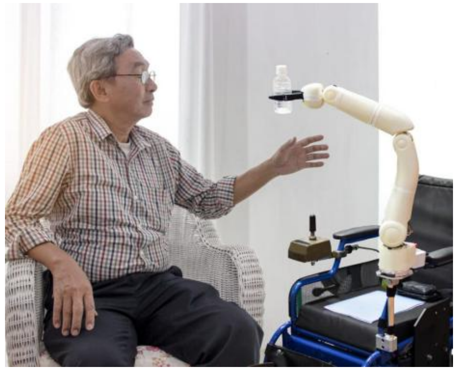
Skills

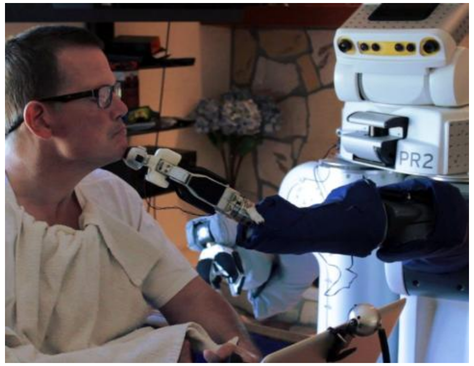
Side to side

throwing

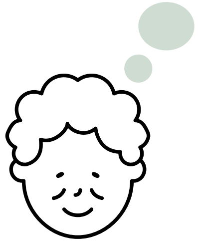
Personalization

Unique individual needs of humans.

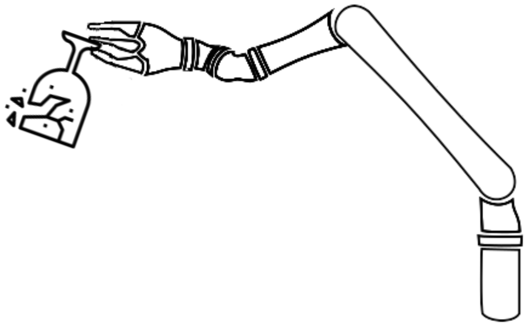




I want the robot to raise **all fragile objects** higher



Train with glassware? electronics? different materials?



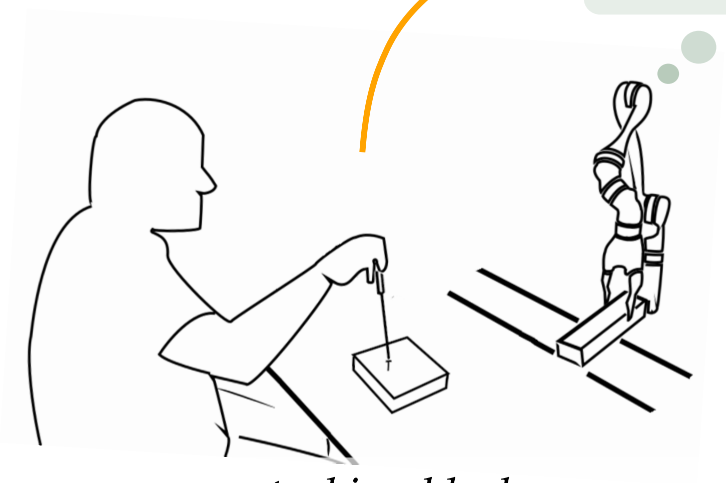
Focus: Choosing a task to learn human *preferences*

Transfer learning

Approach: Transfer objectives learned from demonstrations in simple task to the complex task.

weights of reward function

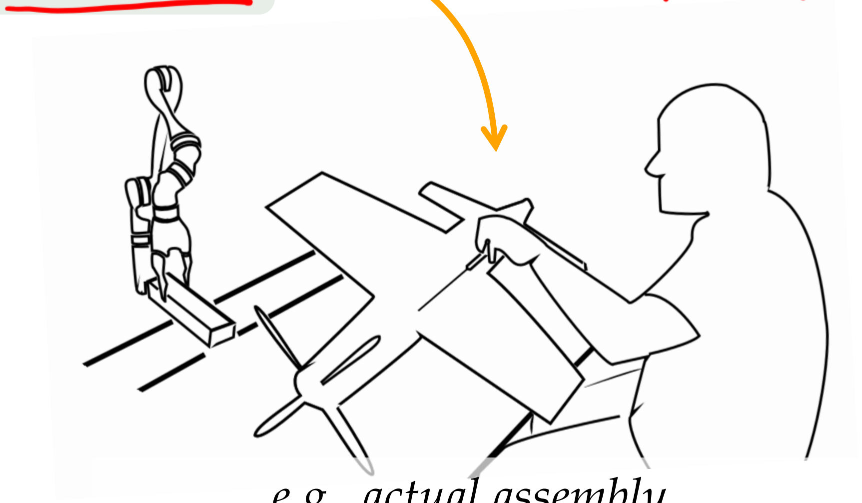
Learn ***task-agnostic*** objectives



e.g., stacking blocks

Users demonstrate simple task

Source



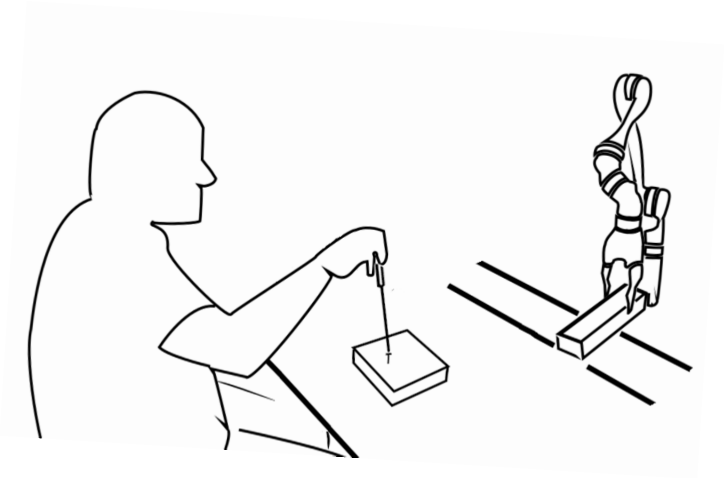
e.g., actual assembly

Robot assists in complex task

Task definition

Model the simple task **(source)** and the complex task **(target)** as MDPs.

Markov Decision Process (MDP): (S, A, T, R)



S - set of states $s_t \in S$

A - set of actions $a_t \in A$

$T(s_{t+1}|s_t, a_t)$ - probability of transitioning to next state $s_{t+1} \in S$

$R(s_{t+1})$ - *reward received by the user* ***(objective)***

Reward learned in the source MDP must also apply to the target MDP.

Task-agnostic objectives

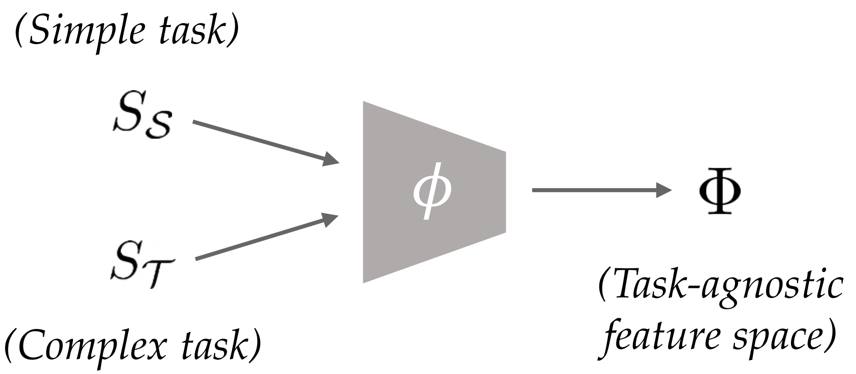
Represent the user's objective as a function of the ***task-agnostic features*** ϕ.

$$R(s) = w^T \phi(s) \quad \forall s \in \{S_{\mathcal{S}}, S_{\mathcal{T}}\}$$

Feature **weights** capture user preferences.

States in the source and target task**.**

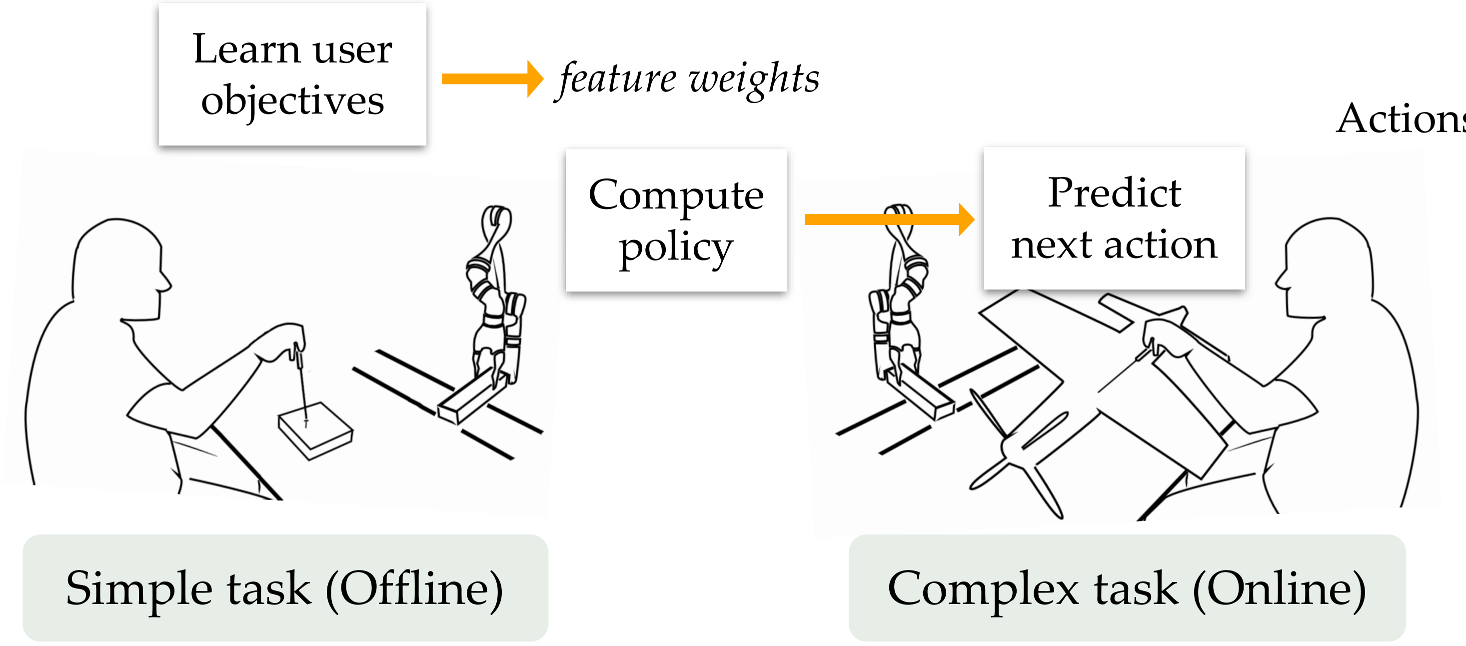
Assembly studies: Features such as ***cost of changing*** parts and tools, ***physical and mental effort*** of actions. *(Fournier et al. 2019, Hesse et al. 2020)*



Transfer learning framework

demonstration

- Learn weights w via ***inverse reinforcement learning*** in source task M_S : $\arg\max_{w} P(w|\xi)$
- Use same weights to compute reward in target task M_T : $R(s) = w^T \phi(s) \quad \forall s \in S_T$.

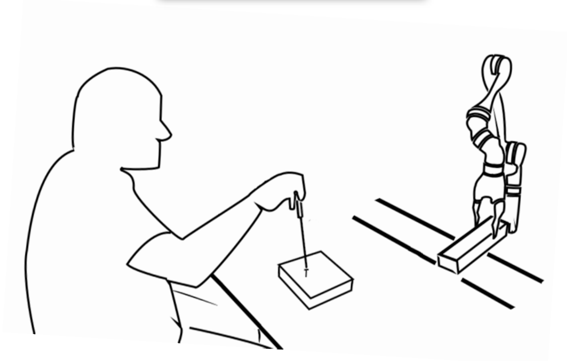


Actions that ***maximizes reward***.

Selecting simple task

- Learn weights w via ***inverse reinforcement learning*** in source task $M_{\mathcal{S}}: \ \arg\max_{w} P(w|\xi)$
- Use same weights to compute reward in target task $M_{\mathcal{T}}: \ R(s) = w^T \phi(s) \quad \forall s \in S_{\mathcal{T}}.$

Learn user objectives



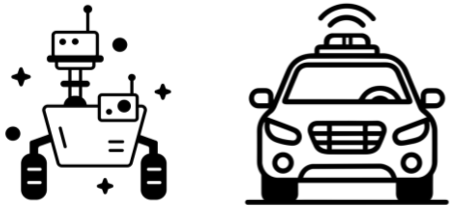
Simple task (Offline)

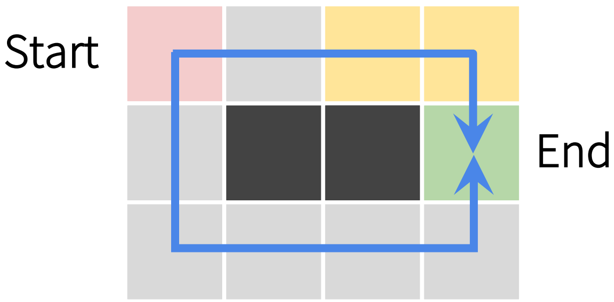
How to ***automatically select simple source tasks*** for transfer learning of human objectives?

Example: Robot navigation

User preferences for spending time and money.

$$R(s) = w_1\phi_1(s) + w_2\phi_2(s)$$

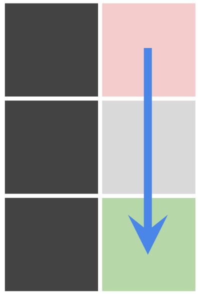




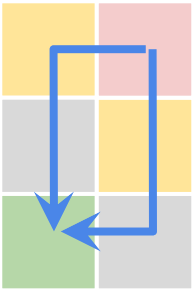
Target task

Time (ϕ_1 = -1)

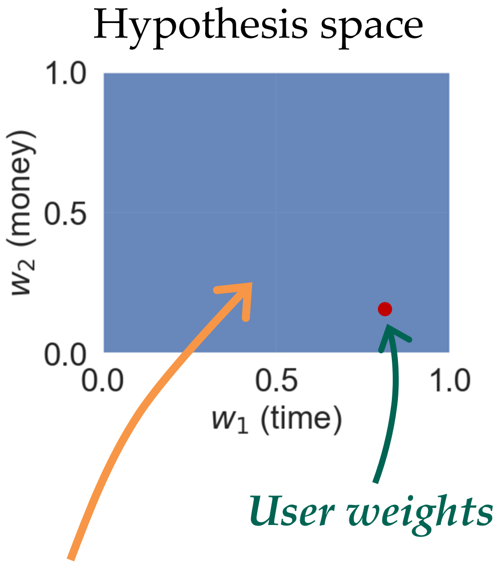
Money (ϕ_2 = -2)



Source A



Source B

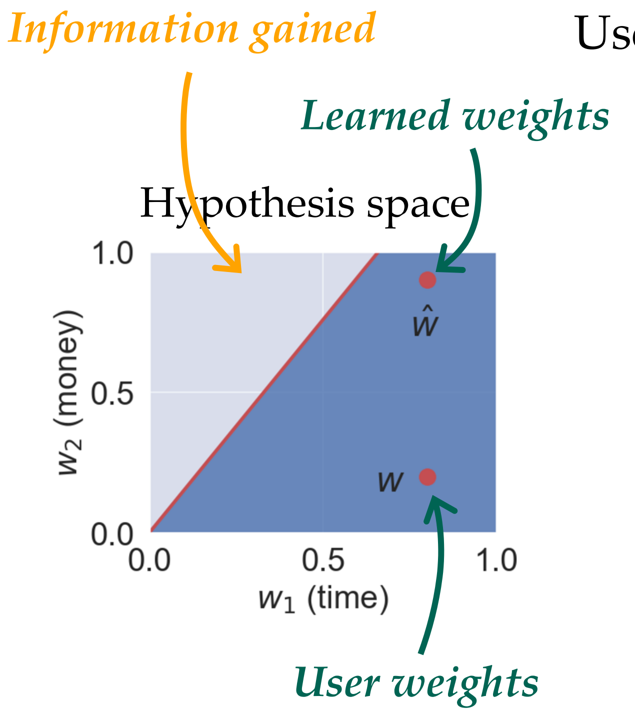


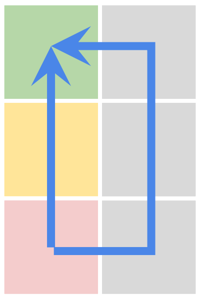
All weights are equally likely. No information gained!

Example: Robot navigation

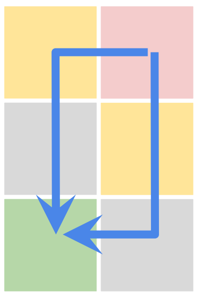
User preferences for spending time and money.

$$R(s) = w_1\phi_1(s) + w_2\phi_2(s)$$

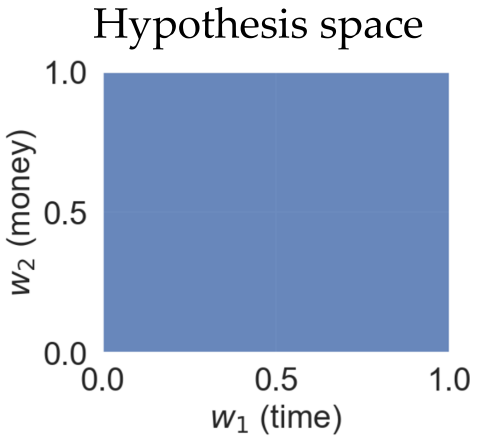




Source C



Source B



Example: Robot navigation

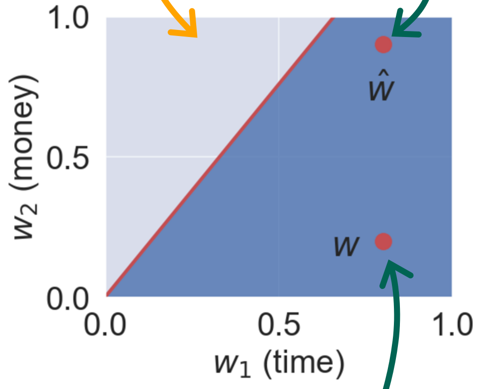
User preferences for spending time and money.

$$R(s) = w_1\phi_1(s) + w_2\phi_2(s)$$

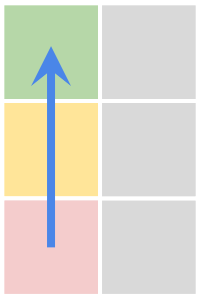
Information gained

Learned weights

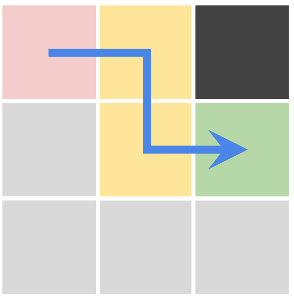
Hypothesis space



User weights



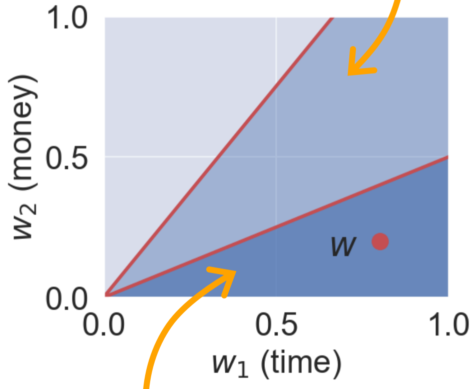
Source C



Source D

More information gained

Hypothesis space



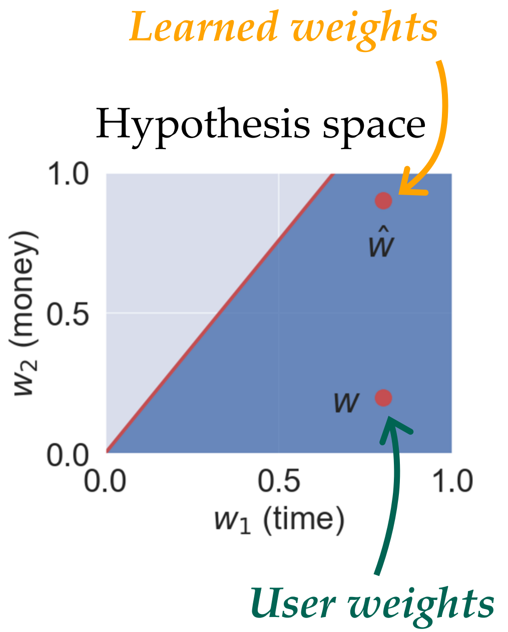
Behavioral equivalence class (BEC)

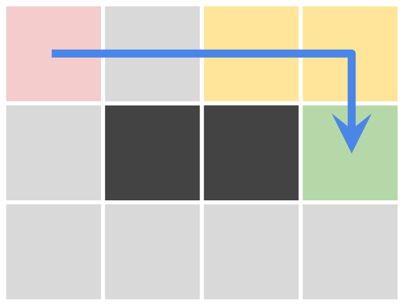
How informative should the source task be?

Example: Robot navigation

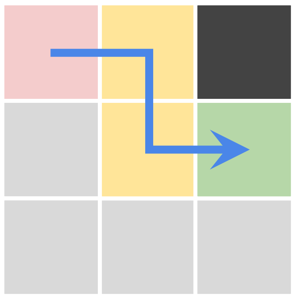
User preferences for spending time and money.

$$R(s) = w_1\phi_1(s) + w_2\phi_2(s)$$

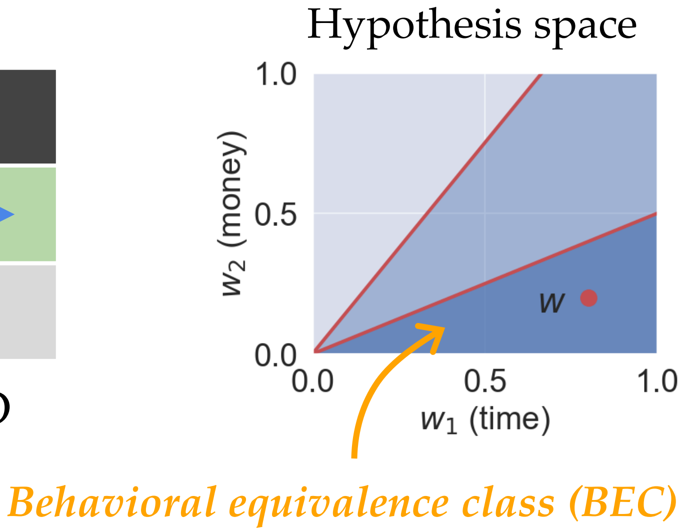




Target task



Source D



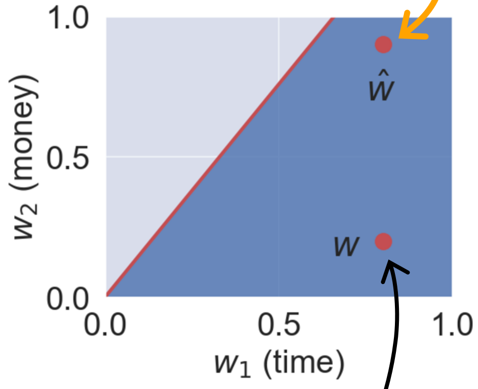
Example: Robot navigation

User preferences for spending time and money.

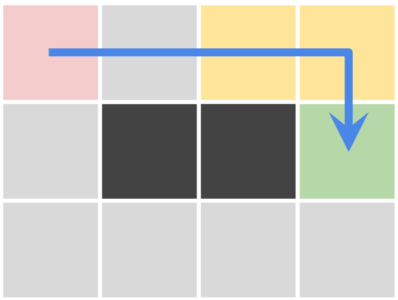
$$R(s) = w_1\phi_1(s) + w_2\phi_2(s)$$

Learned weights

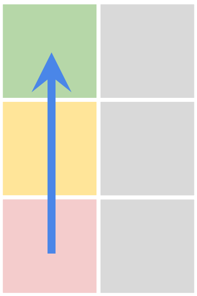
Hypothesis space



User weights

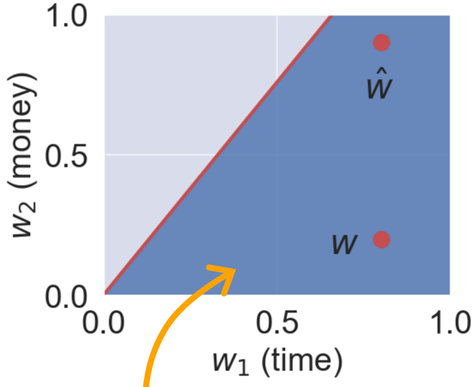


Target task



Source C

Hypothesis space

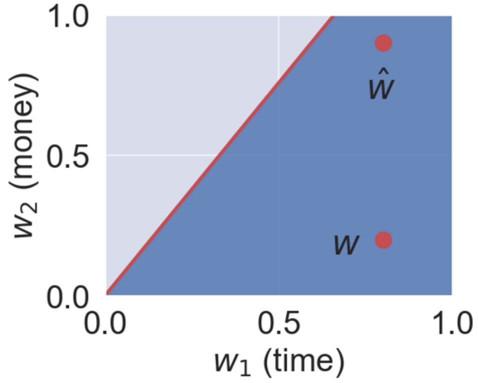


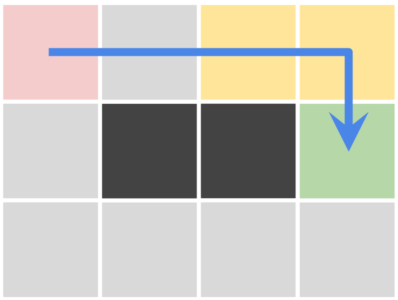
Behavioral equivalence class (BEC)

Insight: Source task only needs to be *behaviorally similar* the target task.

Example: Robot navigation

Hypothesis space

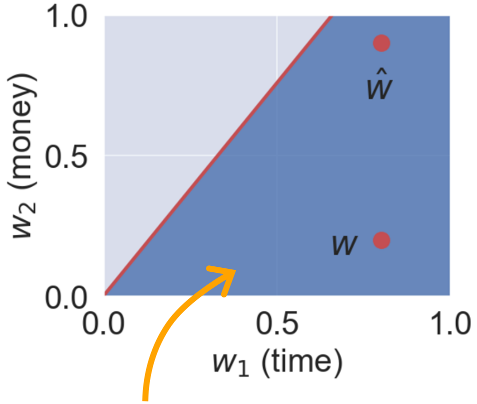




Target task

Source C

Hypothesis space



Behavioral equivalence class (BEC)

Metric: Select source tasks with ***similar behavioral equivalence classes***.

$$BECS(M_{\mathcal{S}}, M_{\mathcal{T}}) = \frac{1}{|W|} \sum_{w_i \in W} P(w_i, M_{\mathcal{S}}, M_{\mathcal{T}})$$

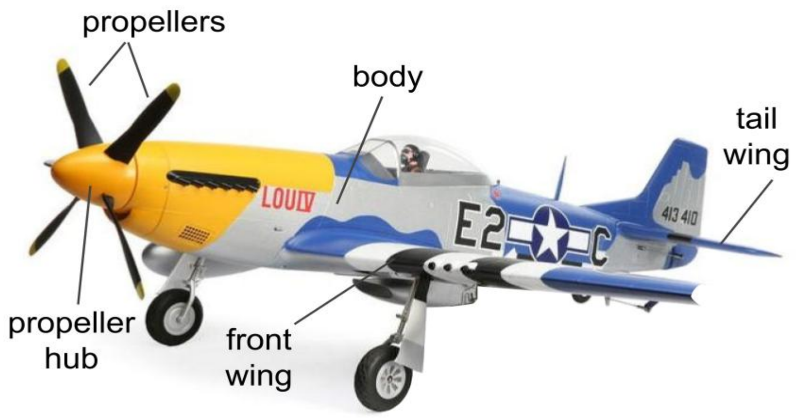
$P(w_i, M_{\mathcal{S}}, M_{\mathcal{T}}) =$ proportion of weights in $BEC(w_i | M_{\mathcal{S}})$ that also belong in $BEC(w_i | M_{\mathcal{T}})$

[source] *[target]*

Human-robot assembly

Robot proactively assists humans without demonstrations in complex assembly task.

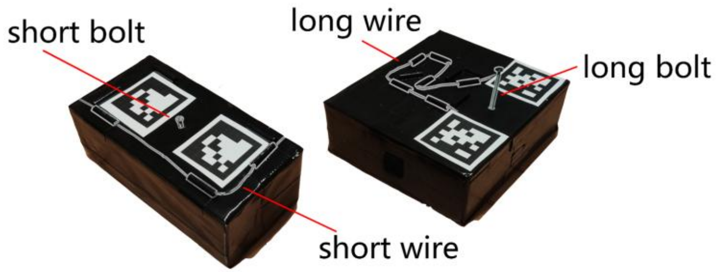
Steps = 17 *Time ≈ 9 min*



Model airplane assembly

1. Procedurally generate several source tasks.
2. Select ***shortest behaviorally similar*** source.

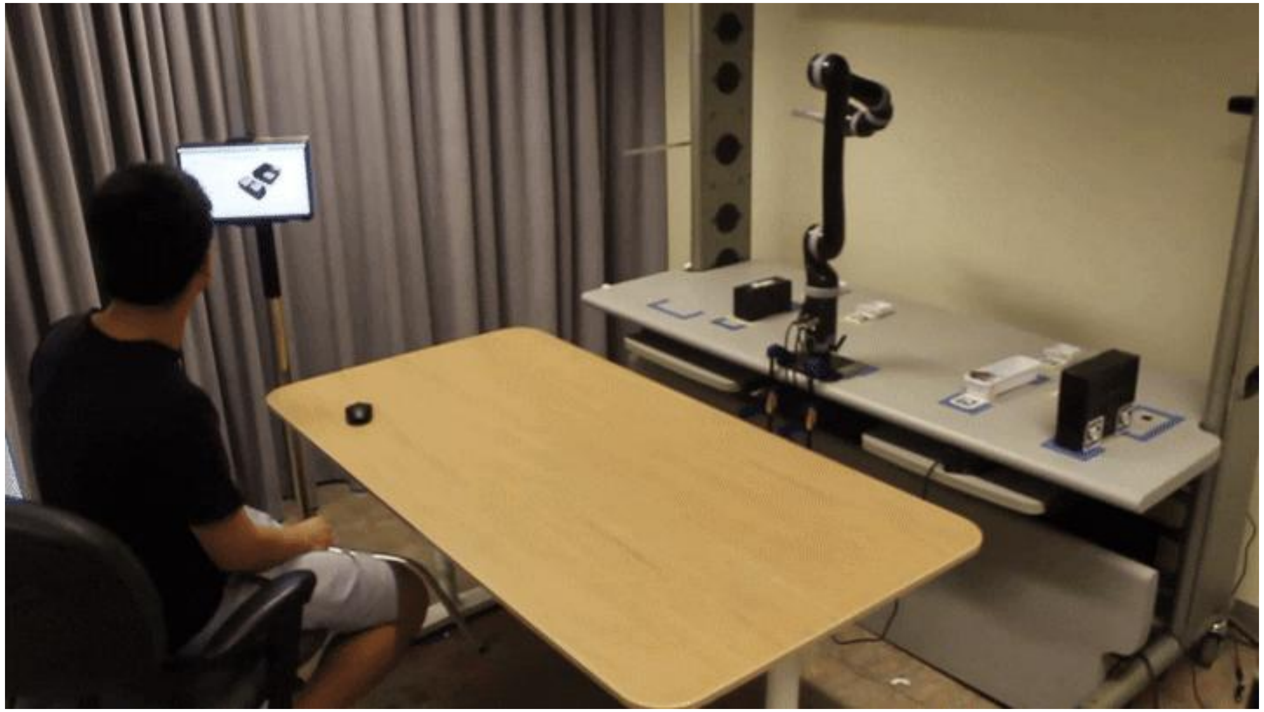
Steps = 6 *Time ≈ 4 min*



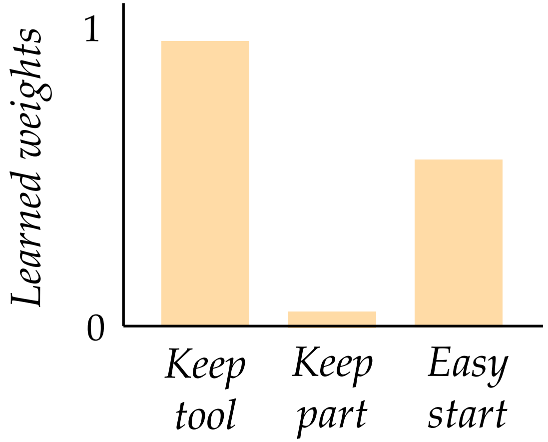
Simple source assembly

Learning in source assembly

- **Human** manually requests required parts.
- **Robot** *reactively* assists user and learns task-agnostic objectives.

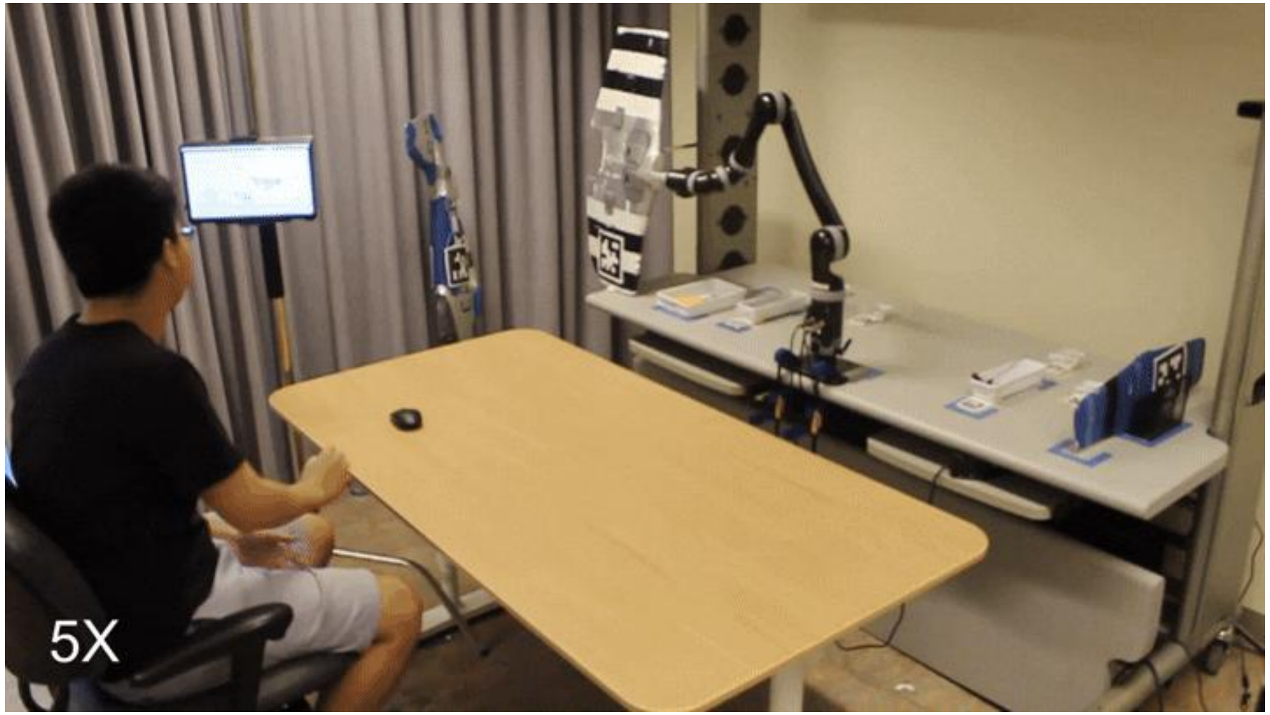


User prefers to ***not switch tools*** *and* ***start with low effort*** *actions.*



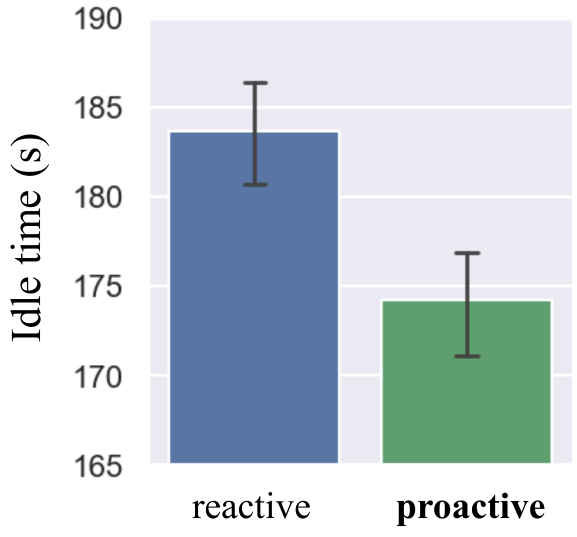
Assisting in target assembly

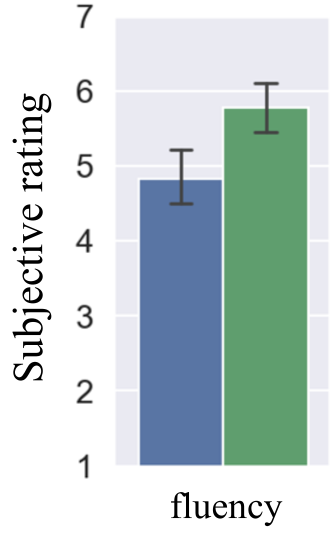
- **Robot** predicts next assembly action and *proactively* reaches required part.
- **Human** provides feedback for online learning.



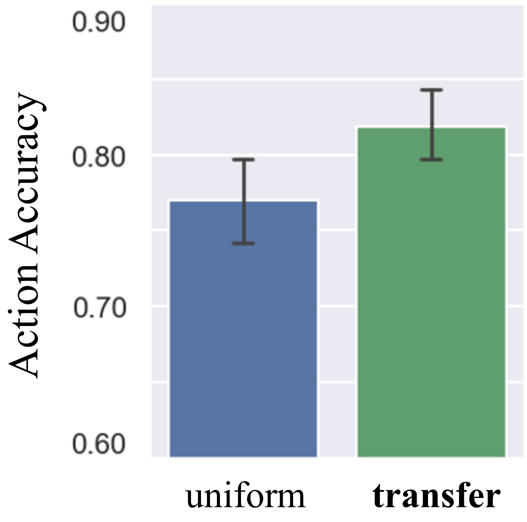
Increasing productivity

Benefit of *proactively assisting* user based on transferred objectives compared to *reactively following* user commands.





Transferred weights improve action accuracy compared to *uniform weights*.



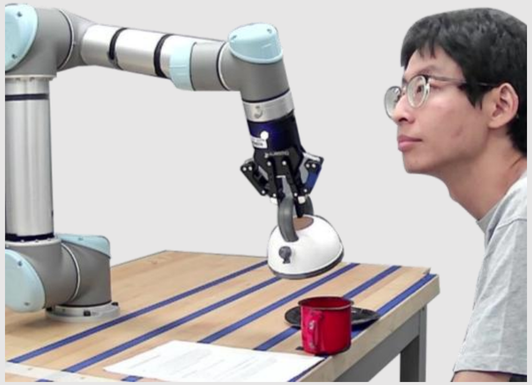
Simplifying robot learning

How can we make it ***easy*** for humans ***to program robots***?

Learning from
simpler tasks

Collaborative assembly

Using intuitive
tools & interfaces



Household tasks

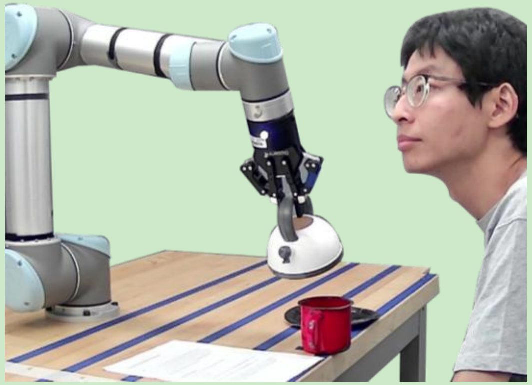
Simplifying robot learning

How can we make it ***easy*** for humans ***to program robots***?

Learning from
simpler tasks

Collaborative assembly

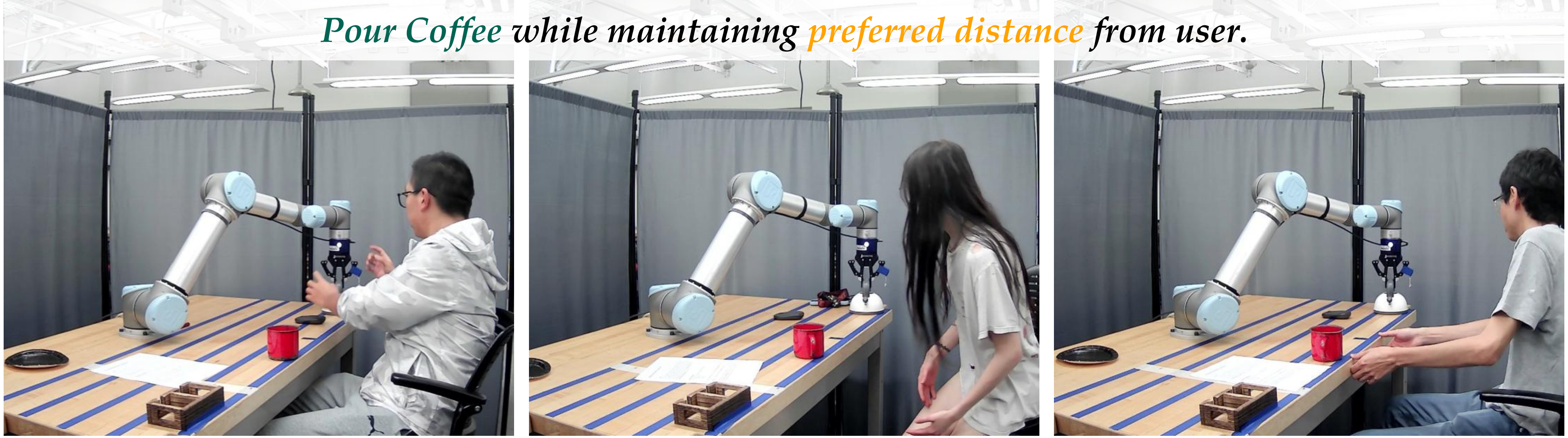
Using intuitive
tools & interfaces



Household tasks

Non-expert users

Challenging for novice end users to demonstrate robot motions accurately.



Multiple household tasks: handover, pick and place, or folding.

Research question

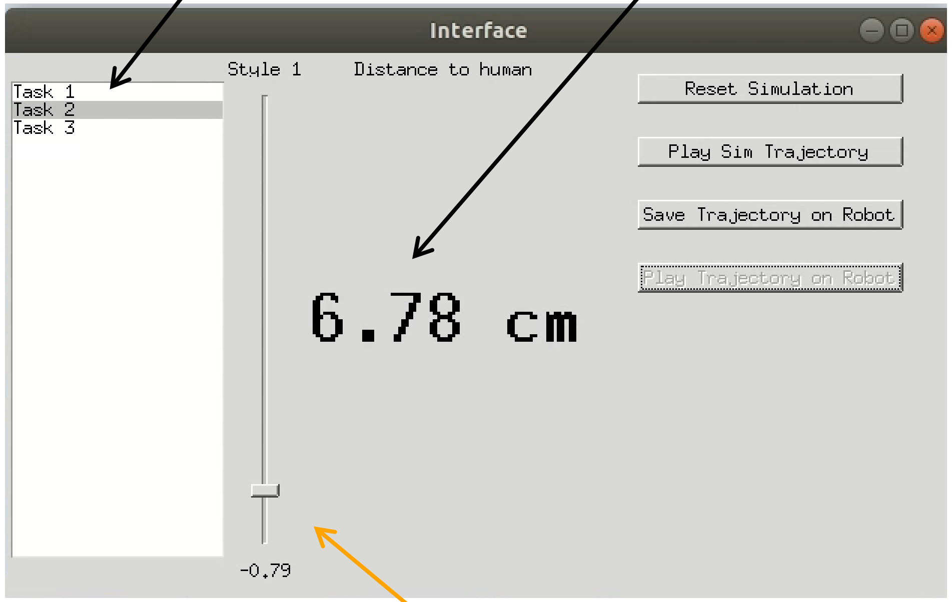
How can users change the robot's behavior without providing demonstrations?

PECAN: Personalizing Robot Behaviors through a Learned Canonical Space (T-HRI 2025)

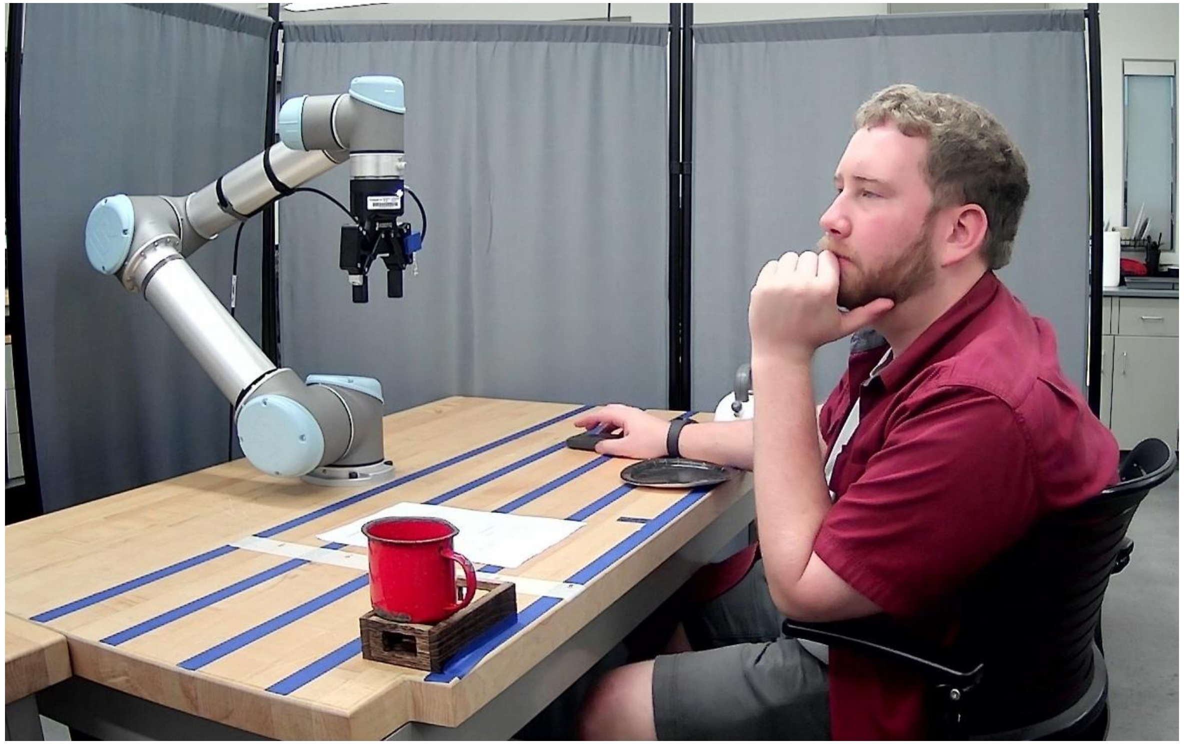
Robot programming interface

End-users select **preferred task** and **robot behavior** from a *low-dimensional representation*.

Task selection **Robot behavior**

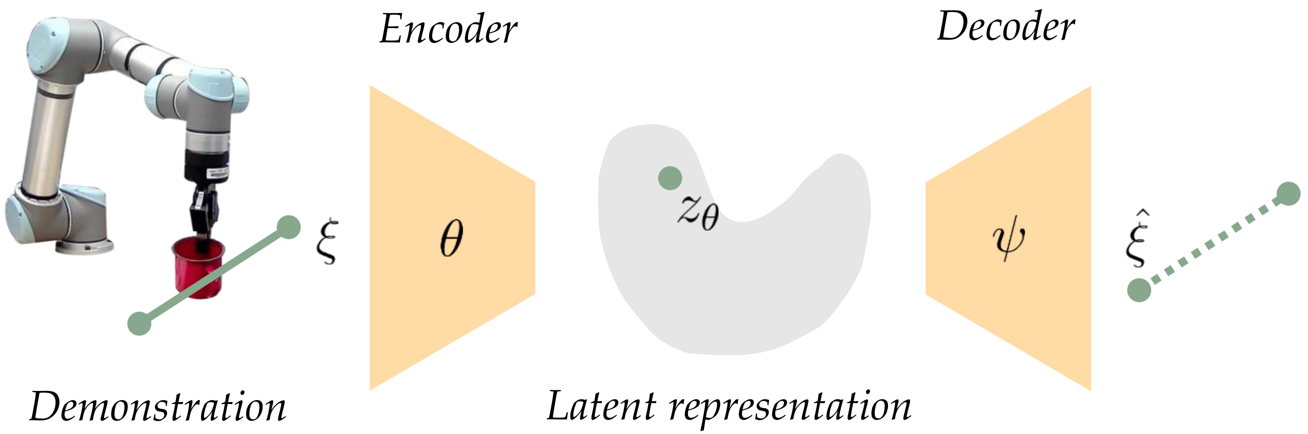


low-dimensional representation



Representation learning

Mapping high-dimensional robot trajectories to low-dimensional representations.

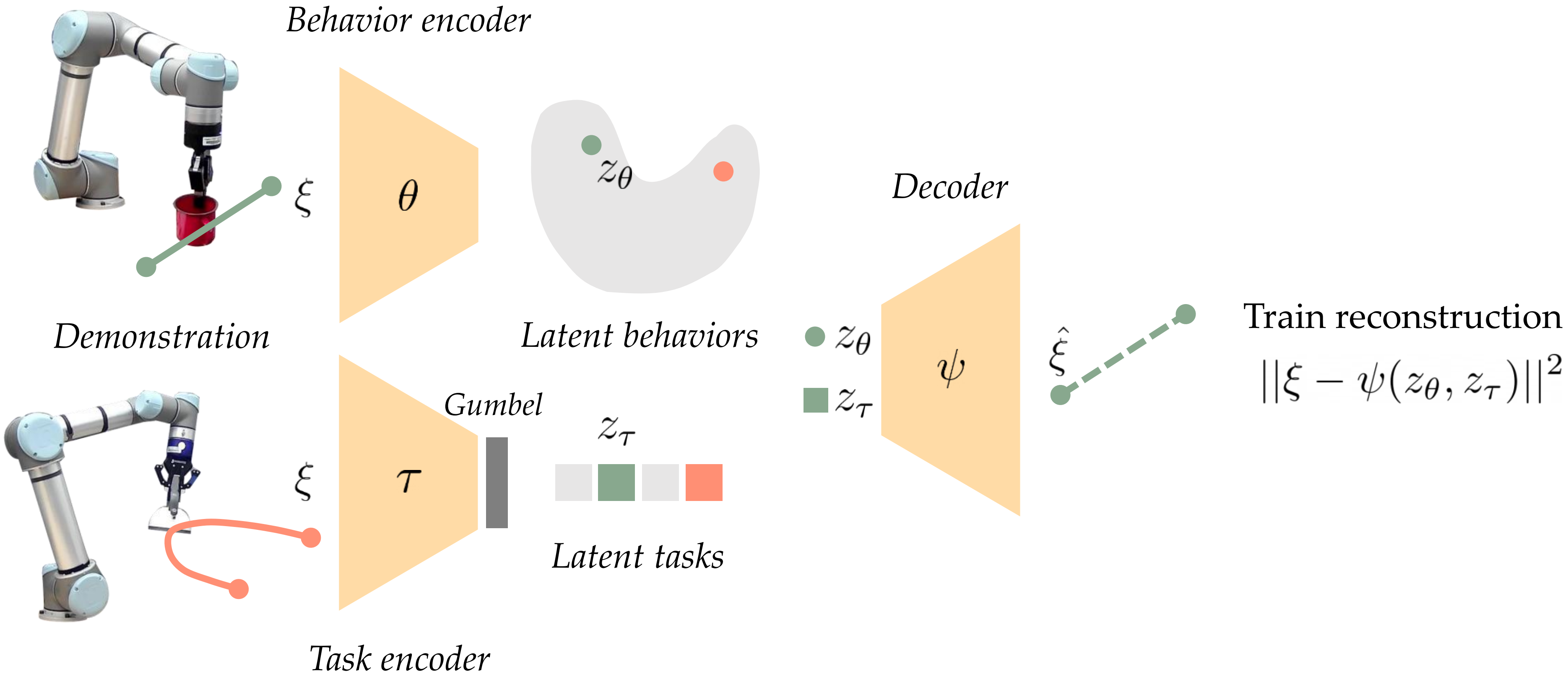


Train reconstruction

$$||\xi - \psi(z_\theta)||^2$$

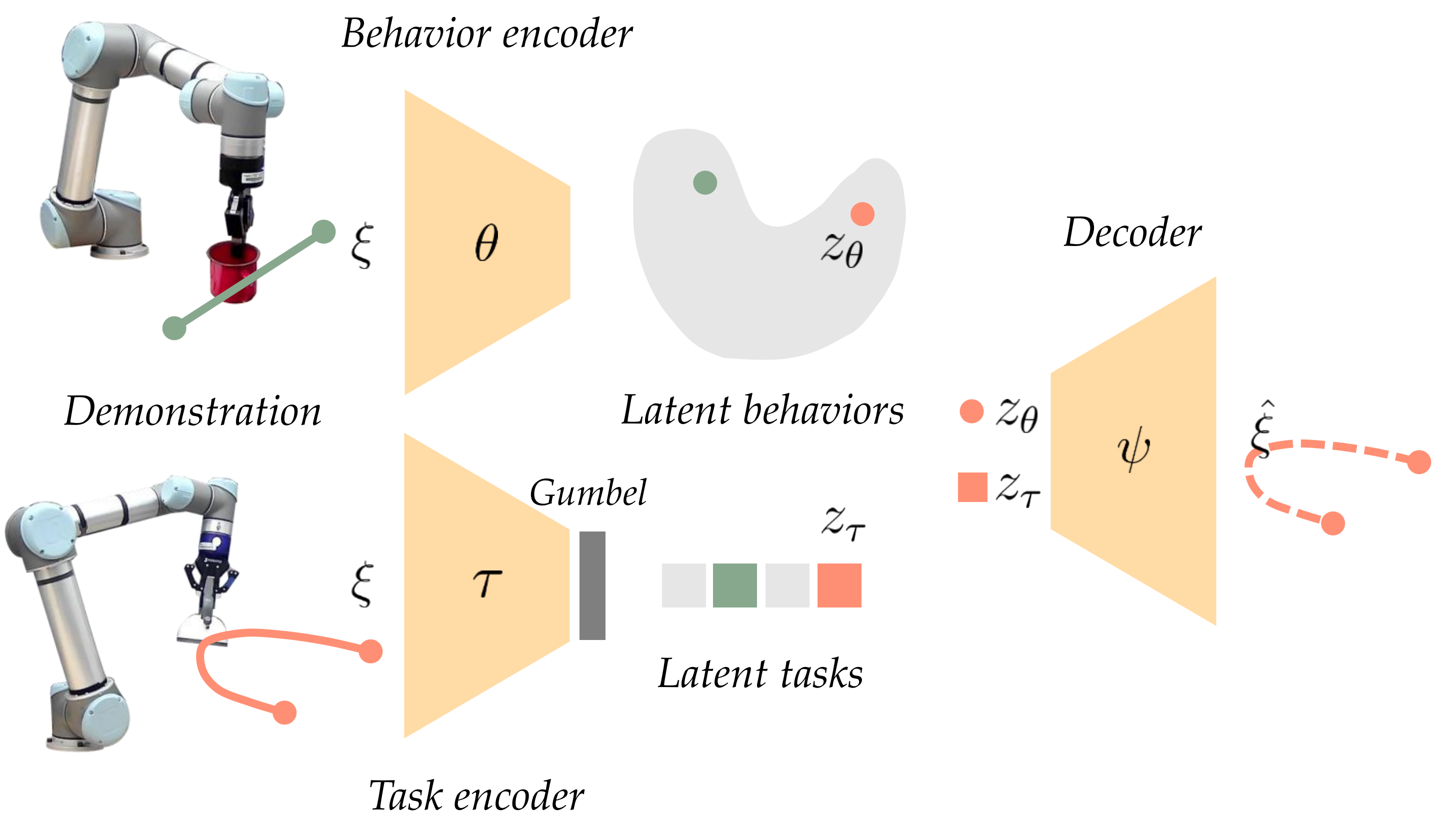
Task representation

Learn *separate latent representations* for tasks and robot behaviors.



Task representation

Learn *separate latent representations* for tasks and robot behaviors.

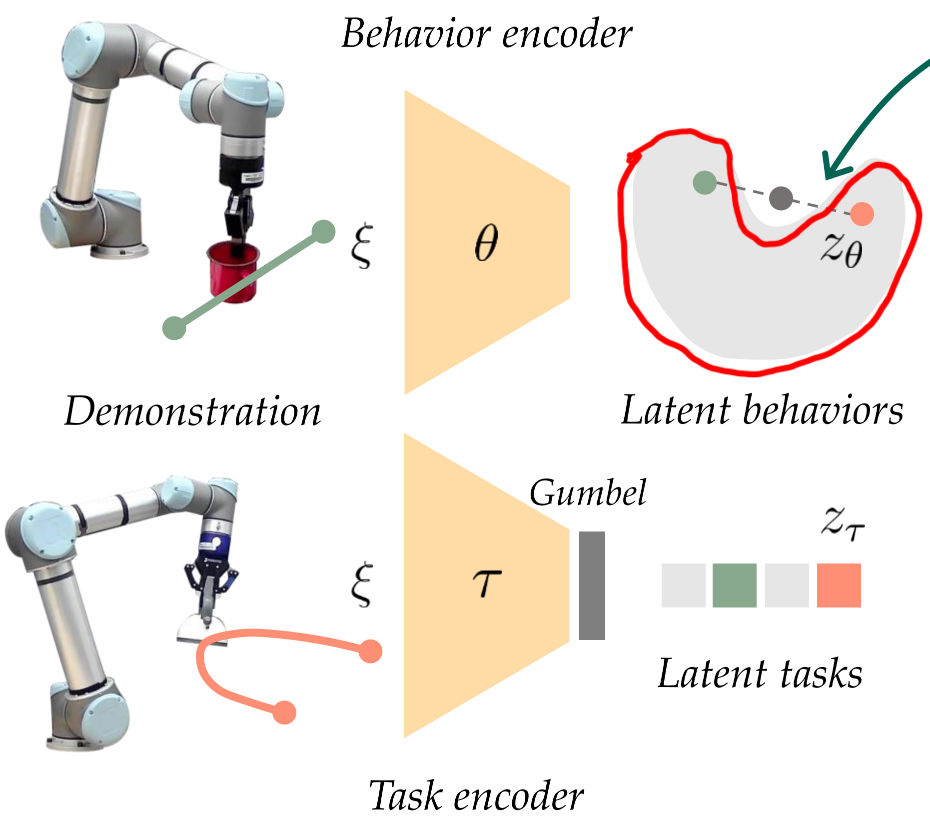


Train reconstruction

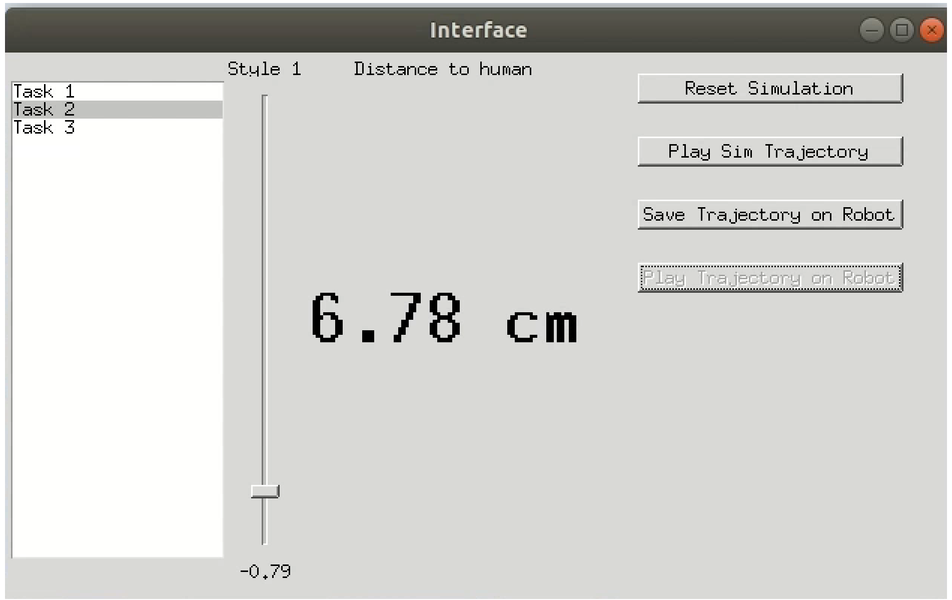
$$||\xi - \psi(z_\theta, z_\tau)||^2$$

Task representation

Learn *separate latent representations* for tasks and robot behaviors.



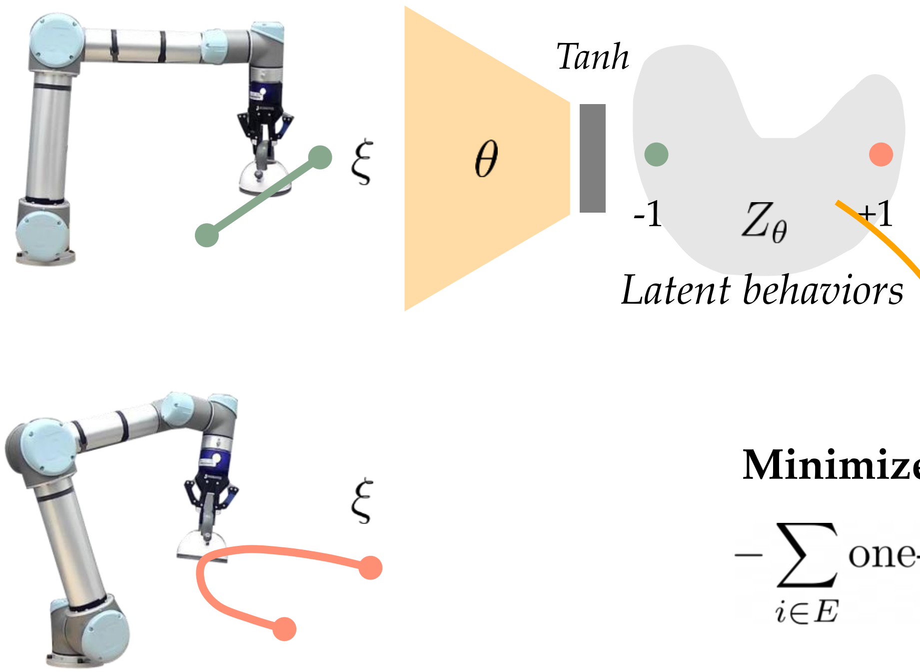
Not user intuitive



Behavior representation

Shape latent space and map ***extreme behaviors*** to opposite ends of latent space.

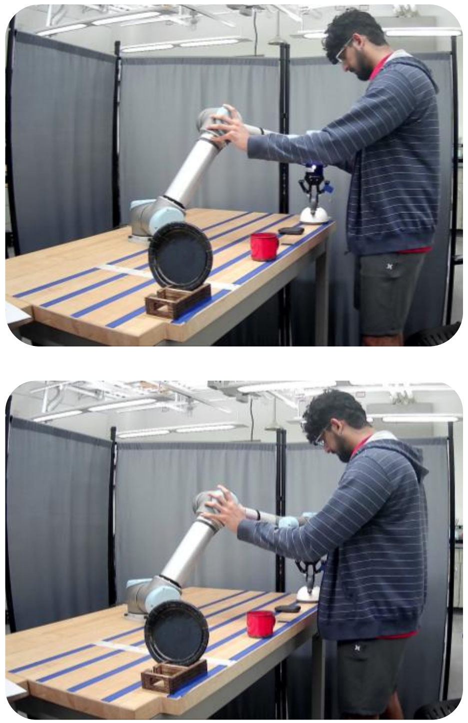
Behavior encoder



Minimize cross-entropy

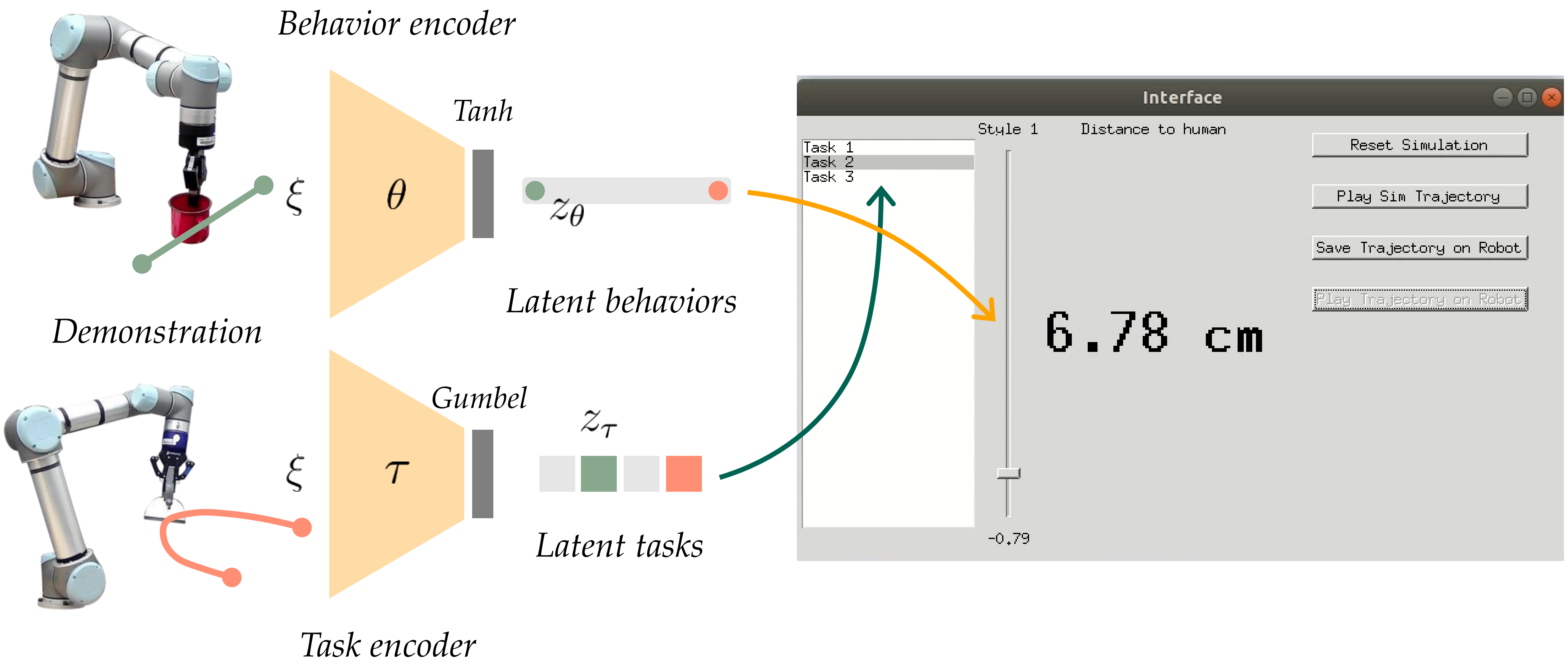
$$-\sum_{i \in E} \text{one-hot}(i) \cdot \log(z_\theta^i)$$

Extreme behaviors E



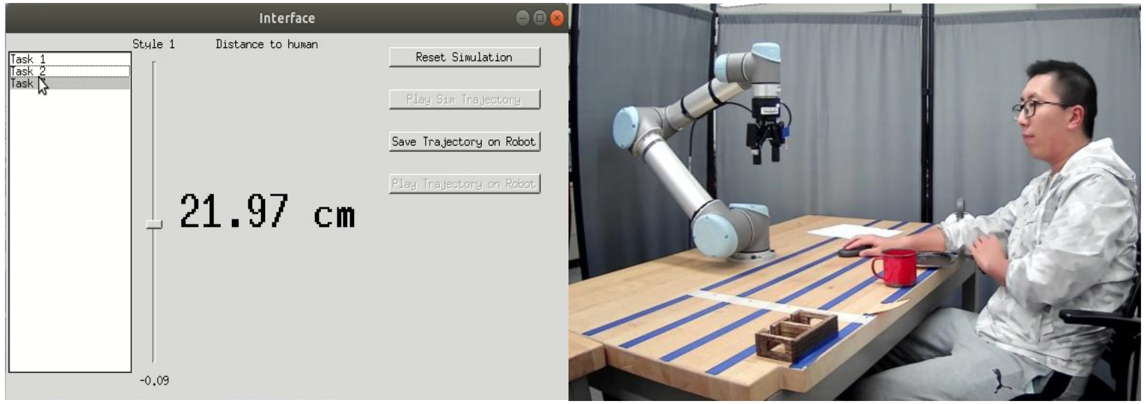
User-friendly interface

Learn *intuitive representations* for easily programming robot behaviors .

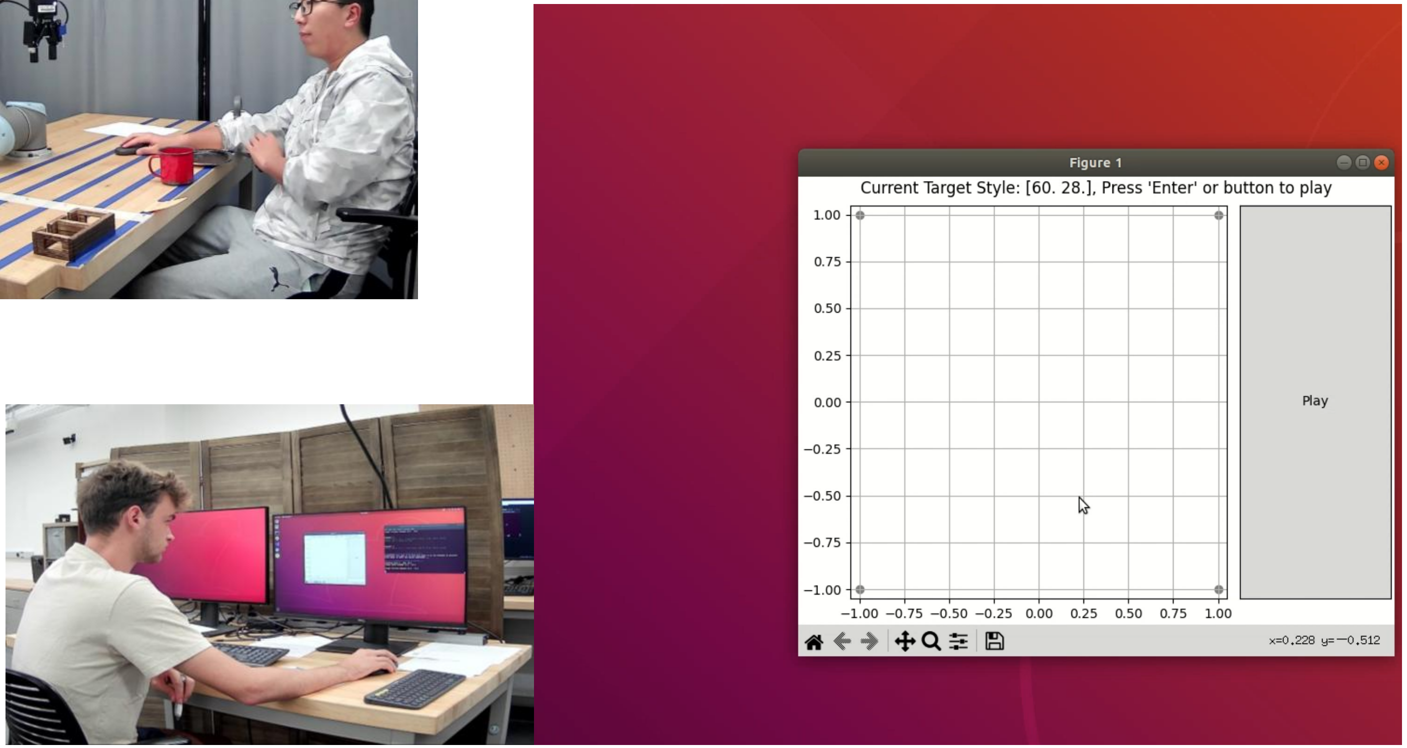


Humans personalizing robots

Robot trajectory proxemics



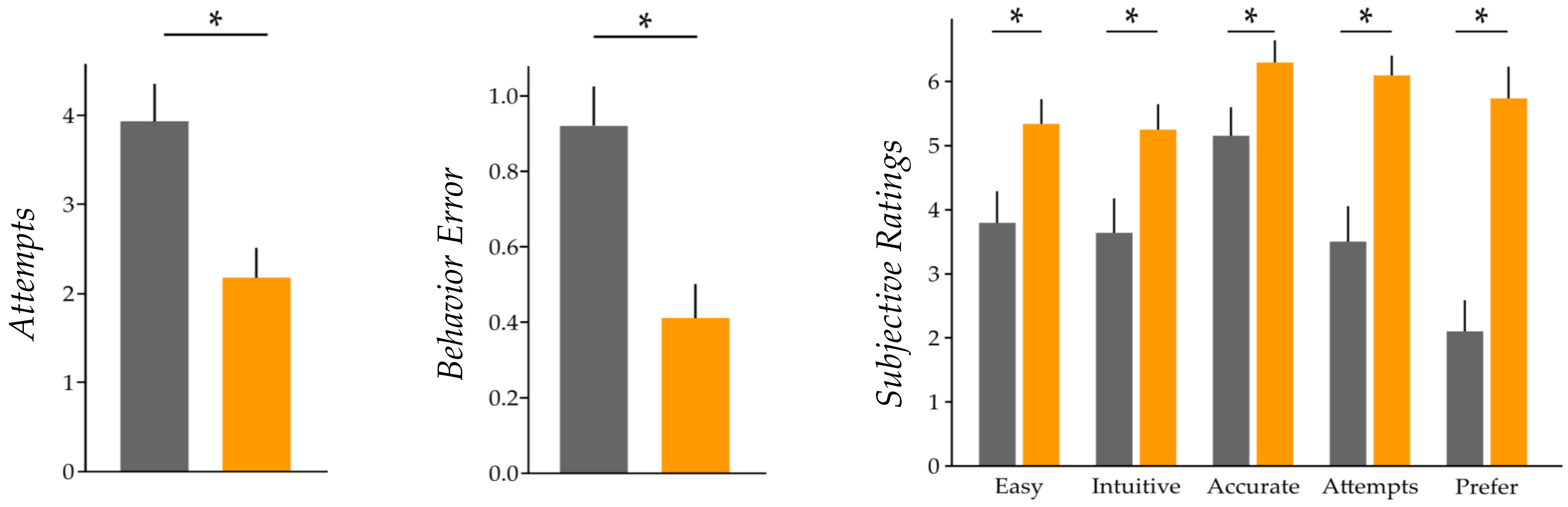
Autonomous driving



Humans personalizing robots

Our ***direct interface*** was more efficient, accurate, and intuitive than ***active learning baseline***.

(Active Preference Learning, Biyik et al. 2022)



Directions

What are some directions of research in imitation learning?

- **Data collection / curation**
  - How data characteristics affect learning performance?
    Towards balanced behavior cloning from imbalanced datasets (arXiv 2025)
  - What data modalities to use? How best to combine them?
    RECON: Reducing causal confusion with human-placed markers (IROS 2025)
    CIVIL: Causal and Intuitive Visual Imitation Learning (arXiv 2025)
- **Efficient usage / learning**
  - How to extract generalizable representations from data?
  - How to learn from data with minimal power consumption?
- ***Is data all you need?***

Simplifying Robot Personalization

Heramb Nemlekar

Assistant Professor

CSUN®

CALIFORNIA
STATE UNIVERSITY
NORTHRIDGE