Teacher-Aware Active Robot Learning

Mattia Racca, Antti Oulasvirta and Ville Kyrki

mattia.racca@aalto.fi

ACM/IEEE International Conference on Human-Robot Interaction (HRI), 2019

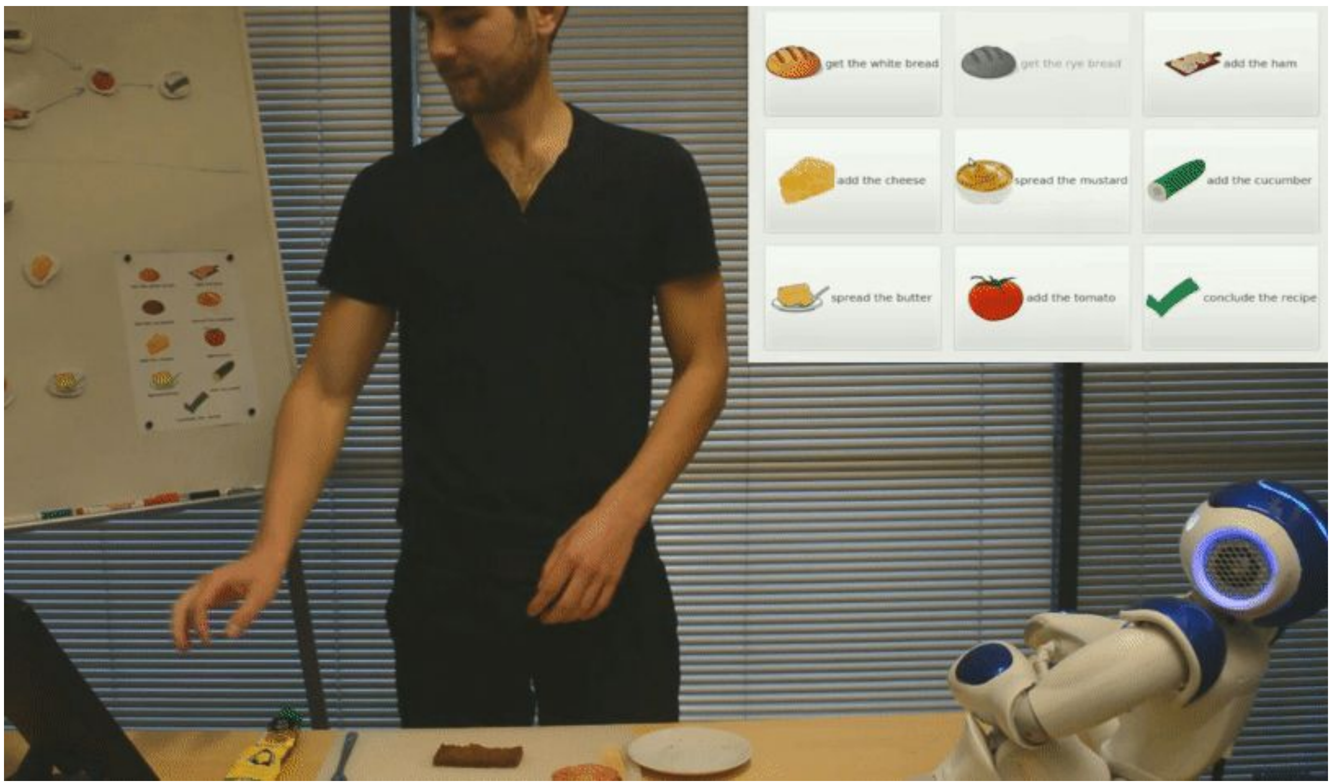
Aalto University
School of Electrical Engineering

Why (active) learning robots?

Programming robots is hard, **pre-programming** them for each task is ~~harder~~ **impossible.**

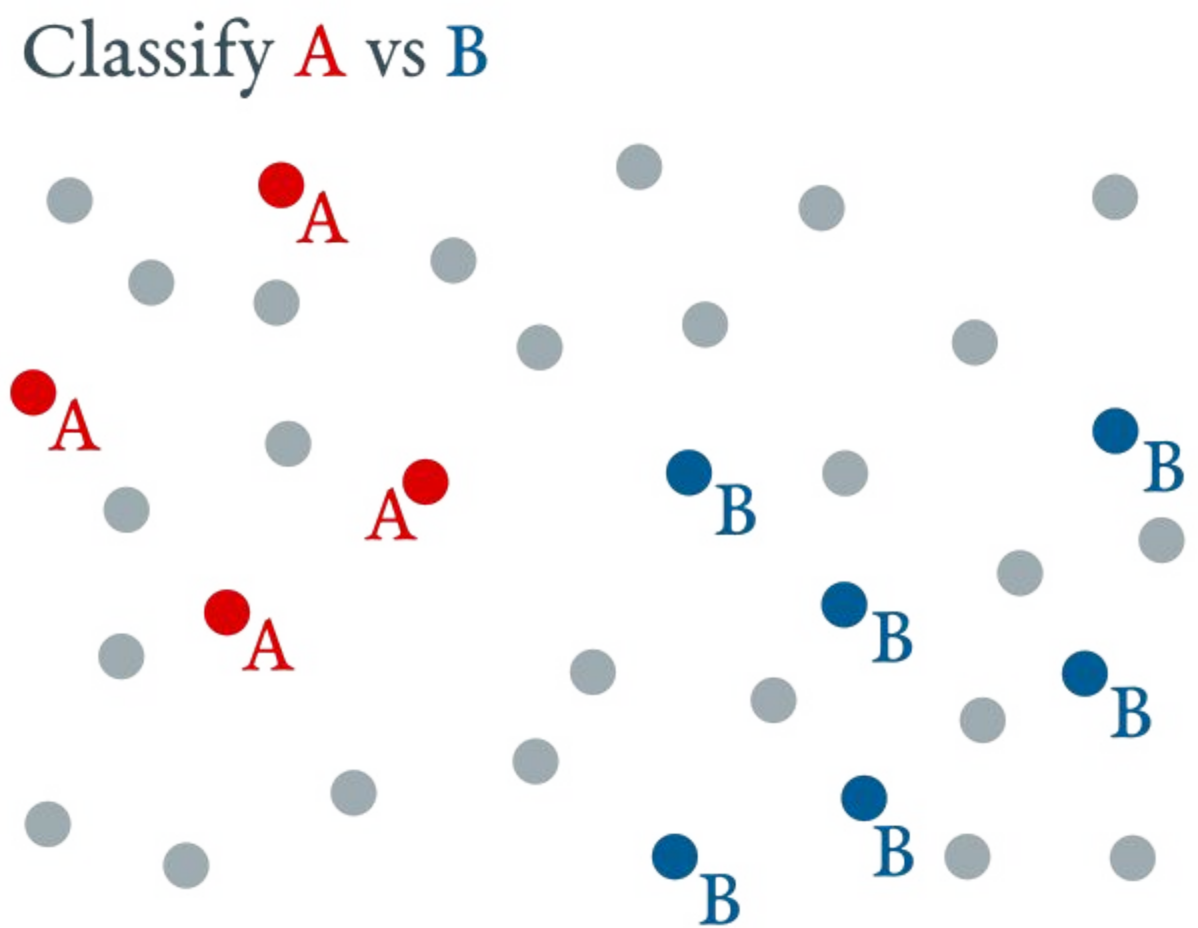
Why (active) learning robots?

Robot should learn by interacting with humans!

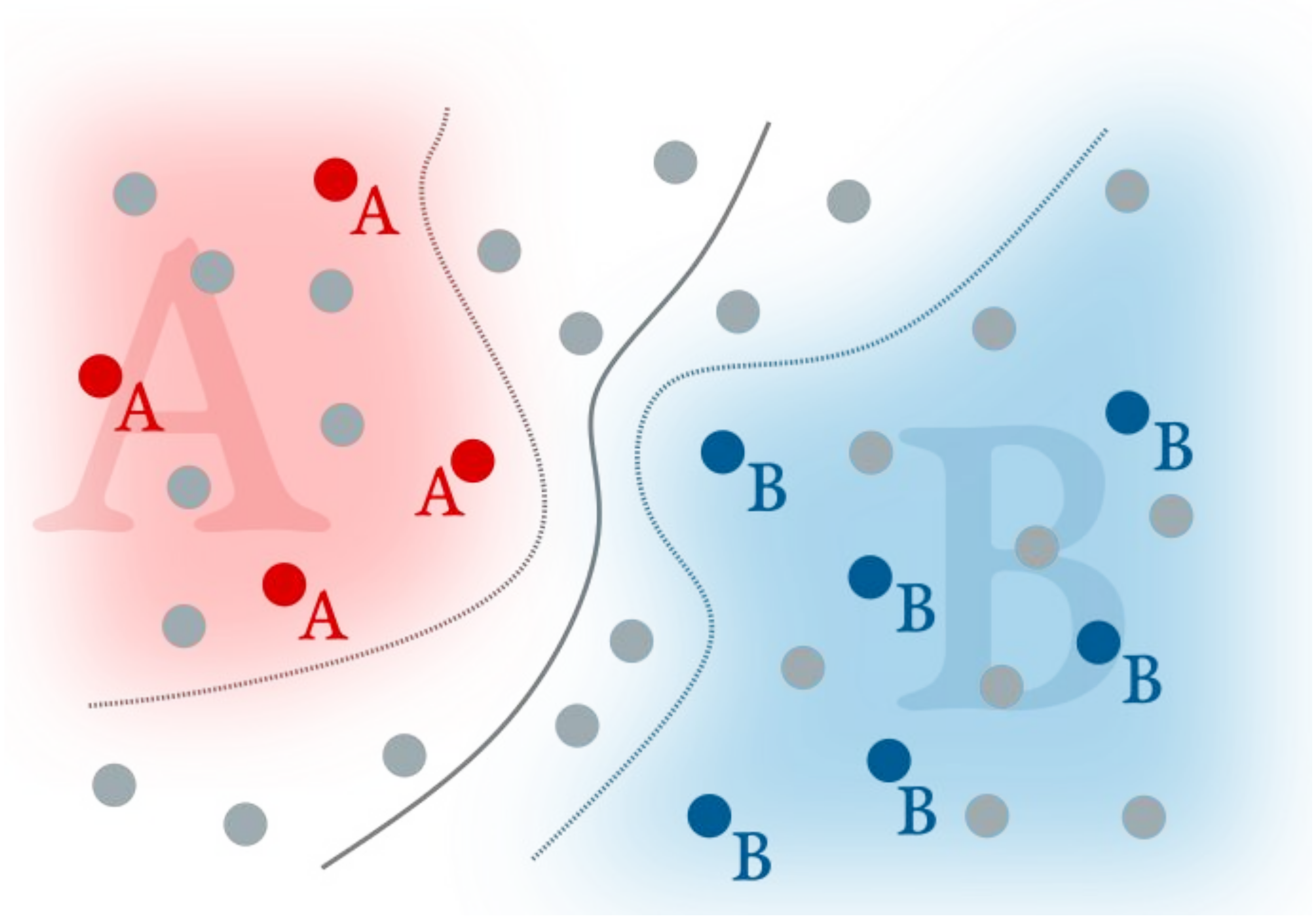


M. Racca and V. Kyrki, **Active Robot Learning for Temporal Task models**, HRI ‘18

The idea behind Active Learning

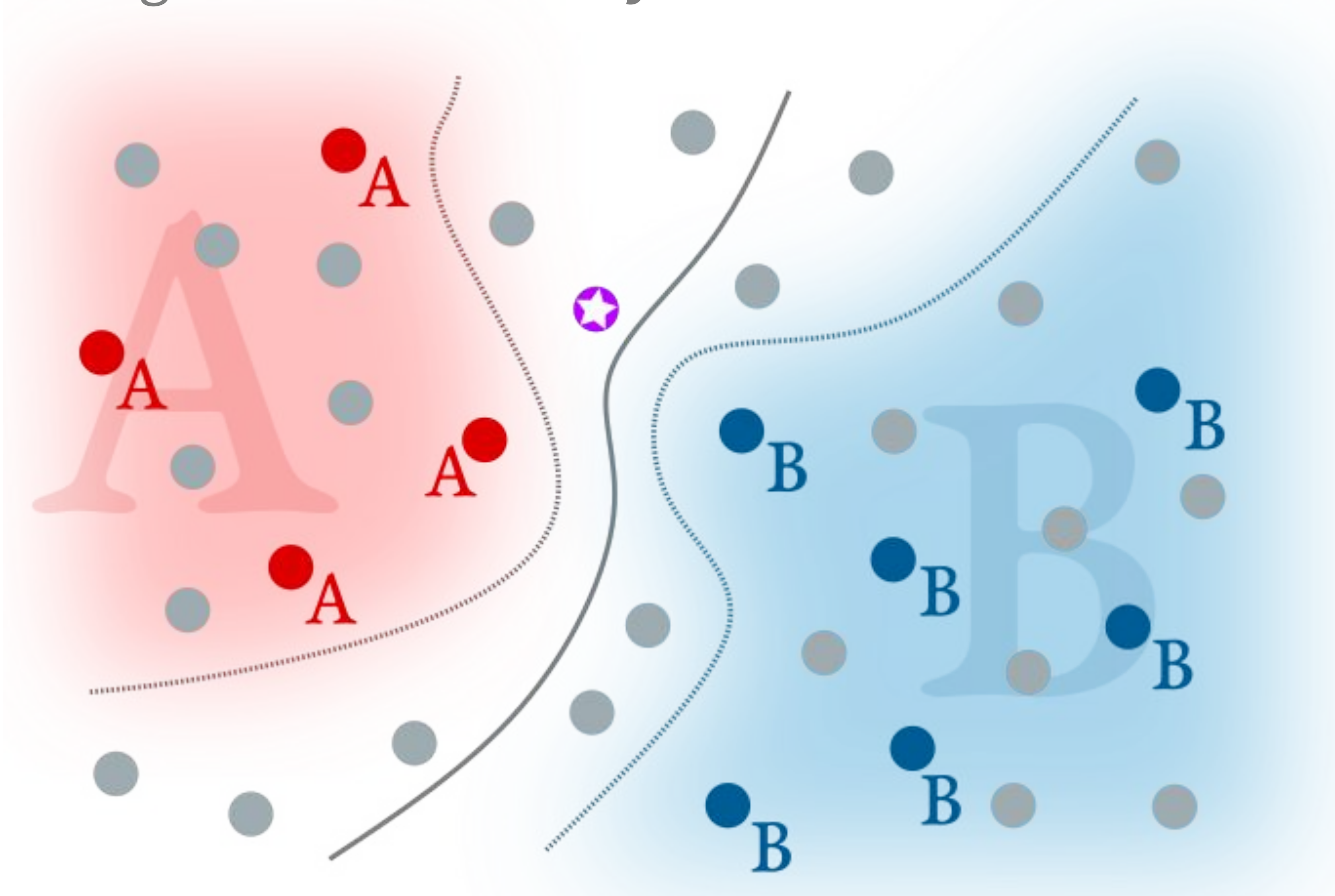


The idea behind Active Learning

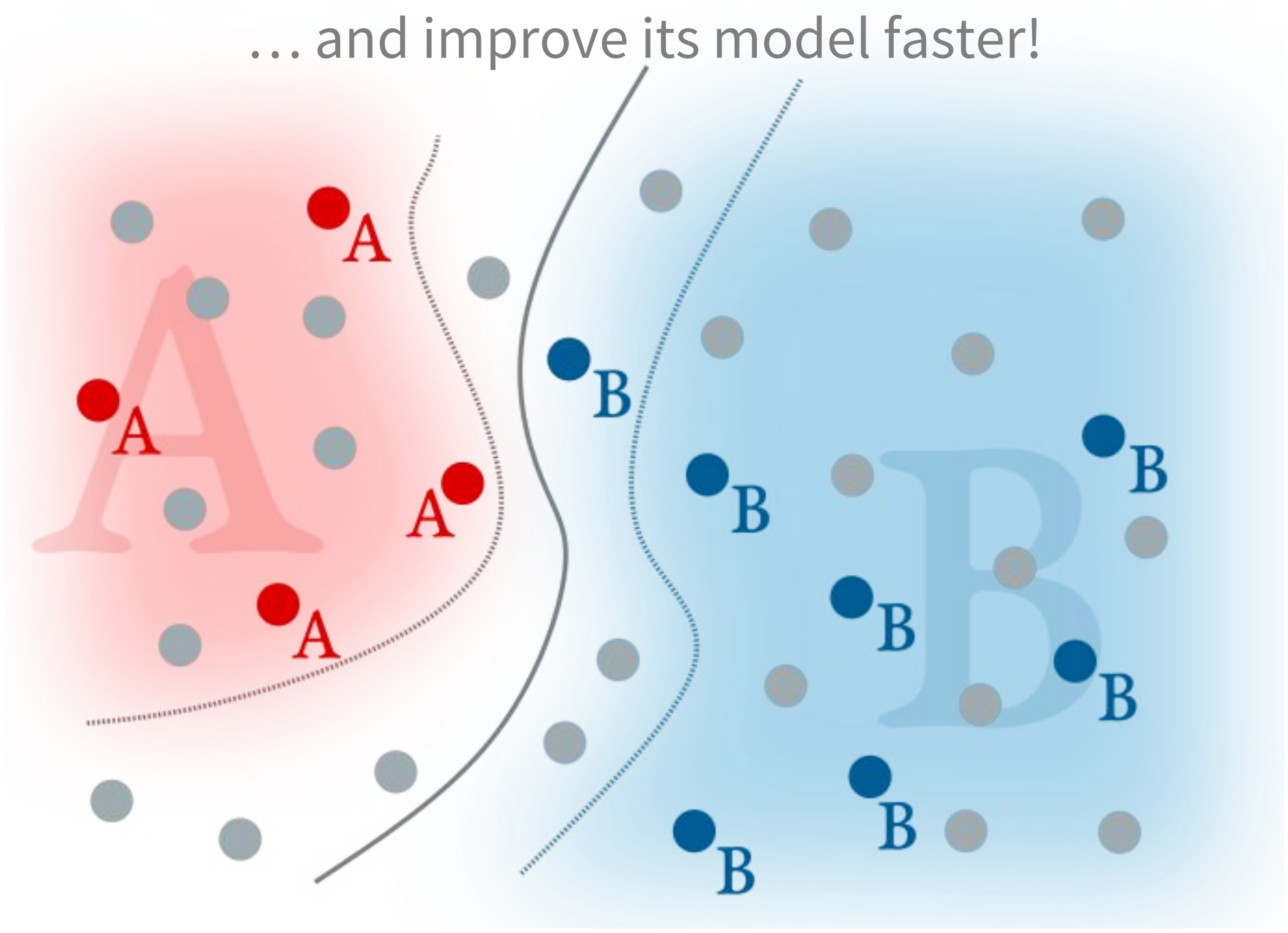


The idea behind Active Learning

The agent can **efficiently** choose what to learn next.



The idea behind Active Learning



Important aspects of Active Learning for HRI

Transparency

Control over interaction

1. **Interactive Nature**

Design of questions

Timing of questions

Important aspects of Active Learning for HRI

Transparency

Control over interaction

1. Interactive Nature

Design of questions

Timing of questions

2. Query Efficiency

Learning faster (with less data)

Important aspects of Active Learning for HRI

Transparency

Control over interaction

1. Interactive Nature

Design of questions

Timing of questions

2. Query Efficiency

Learning faster (with less data)

But what about REAL users?

What if efficient query selection is not best for the interaction?

Query Efficiency

Complex questions

Questions out of context

Harder for the teacher

- slower interaction
- more effort
- more errors!

Can efficiency indirectly counter its own benefits?

Different types of Active Learning

1. CLASSIC AL STRATEGY (LEARNER C)

2. TEACHER-AWARE AL STRATEGY (LEARNER M)

3. HYBRID AL STRATEGY (LEARNER H)

Problem statement & Evaluation scenario

An agent has to **learn the value of a certain attribute** *a* for a set $\mathcal{E}$ of entities by making **queries**. We used the **Animals with Attributes 2*** dataset with 50 animals (entities) and 85 semantic attributes.

* *Y. Xian, et al.*. **Zero-Shot Learning - A Comprehensive Evaluation of the Good, the Bad and the Ugly**, T-PAMI

Problem statement & Evaluation scenario

An agent has to **learn the value of a certain attribute** *a* for a set $\mathcal{E}$ of entities by making **queries**. We used the **Animals with Attributes 2*** dataset with 50 ani semantic attributes.

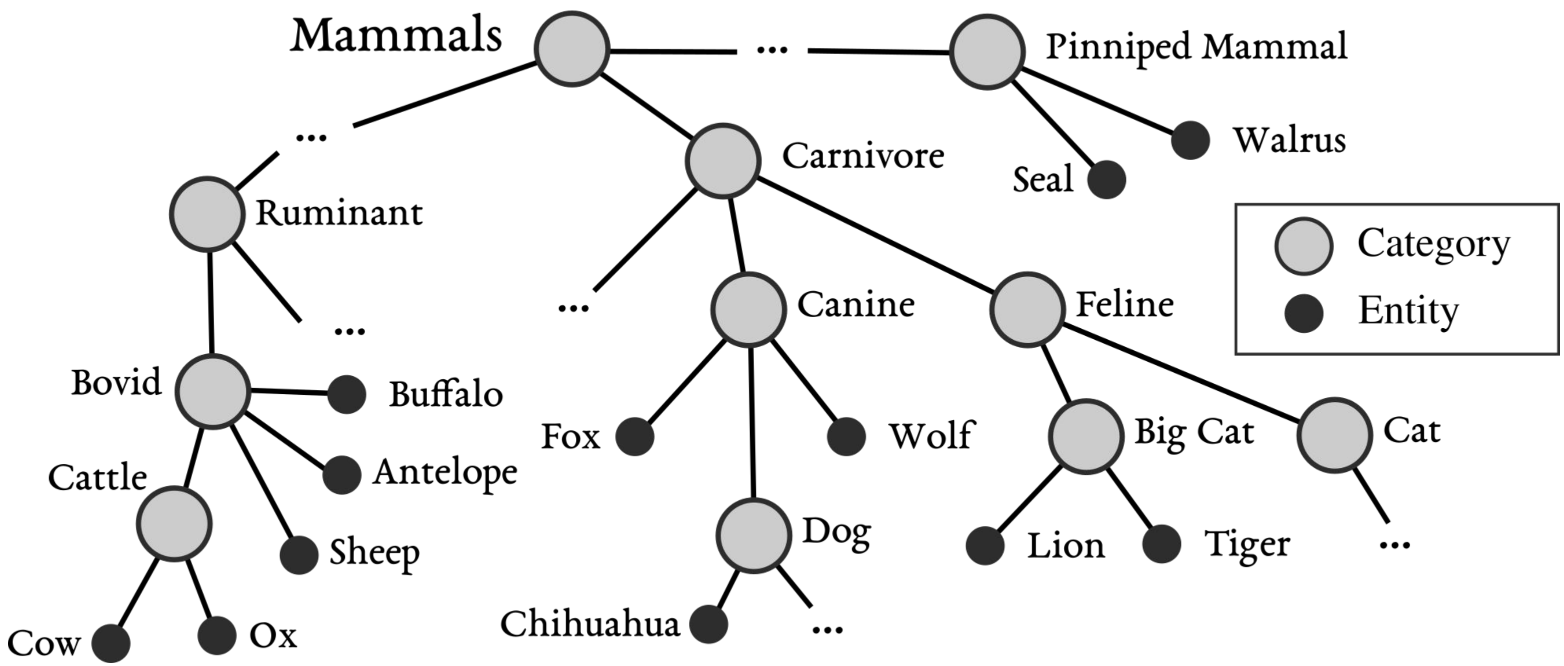
* *Y. Xian, et al.*. **Zero-Shot Learning - A Comprehensive Evaluation of the Good, the Bad and the Ugly**, T-PAMI

Problem statement & Evaluation scenario

- categories $\mathcal{C}$ over entities using **WordNet**
- Learner assumption: ***Entities in the same category are more likely to share the same attribute value.***

Problem statement & Evaluation scenario

- categories $\mathcal{C}$ over entities using **WordNet**
- Learner assumption: ***Entities in the same category are more likely to share the same attribute value.***

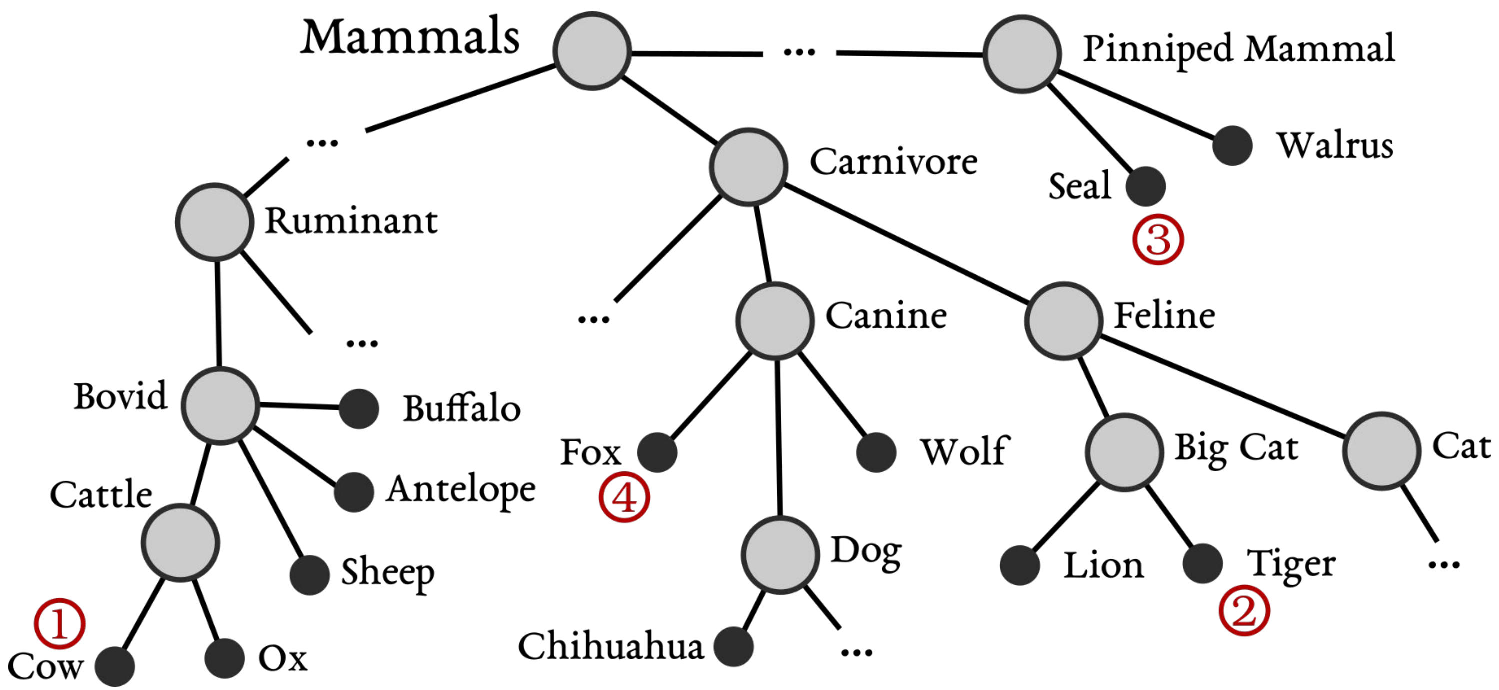


Classic AL: Uncertainty Sampling

- **Learner C**:
  - uses **Uncertainty Sampling**
  - selects the **most uncertain** query, given the current model.
  - As expected **efficient!**

Classic AL: Uncertainty Sampling

- **Learner C drawbacks**
  - Some questions are **difficult!**
  - Topic or **context switches!**



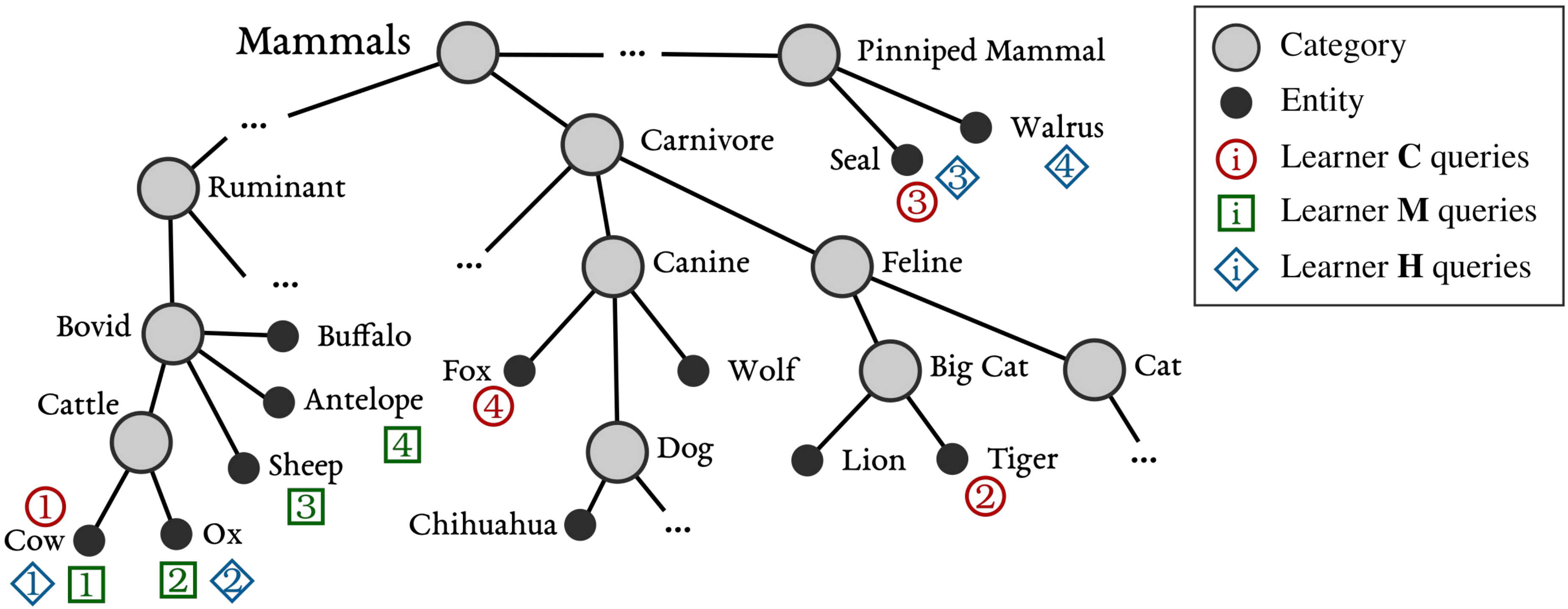
In response to the drawbacks

- **Teacher-Aware** strategy (**Learner M**)
  - Inspired by ACT-R declarative memory model, saying *“Information associated with recently retrieved information is easier to retrieve”*,
  - **minimize the distance between consecutive queries**

In response to the drawbacks

- **Teacher-Aware** strategy (**Learner M**)
  - Inspired by ACT-R declarative memory model, saying *"Information associated with recently retrieved information is easier to retrieve"*,
  - **minimize the distance between consecutive queries**;
- **Hybrid** strategy (**Learner H**)
  - a tradeoff between **Learner C** and **Learner M**

Teacher-Aware AL: Memory Effort strategy

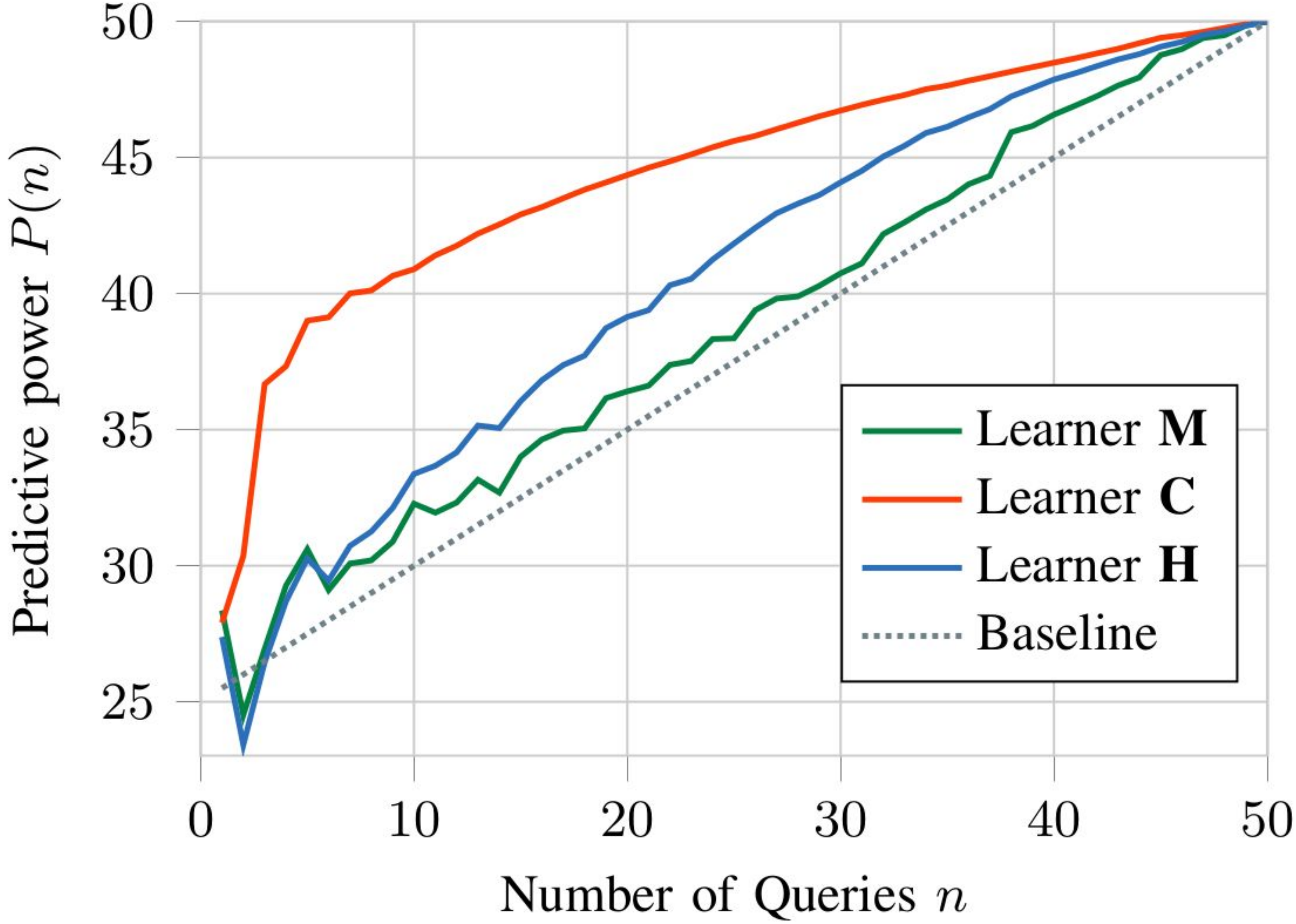


Performance in Simulation

Simulation on the entire dataset:

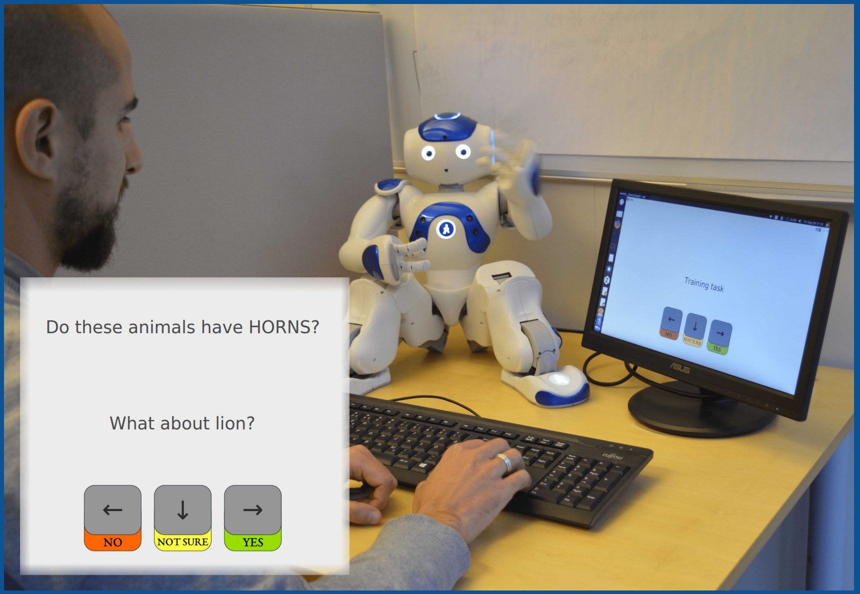
- **Perfect users (no errors, no distraction)**
- **Baseline**: asks random questions and cannot leverage our model to make predictions

Performance in Simulation



What about real users?

User study: 26 participants, the 3 strategies as conditions (within-subject).

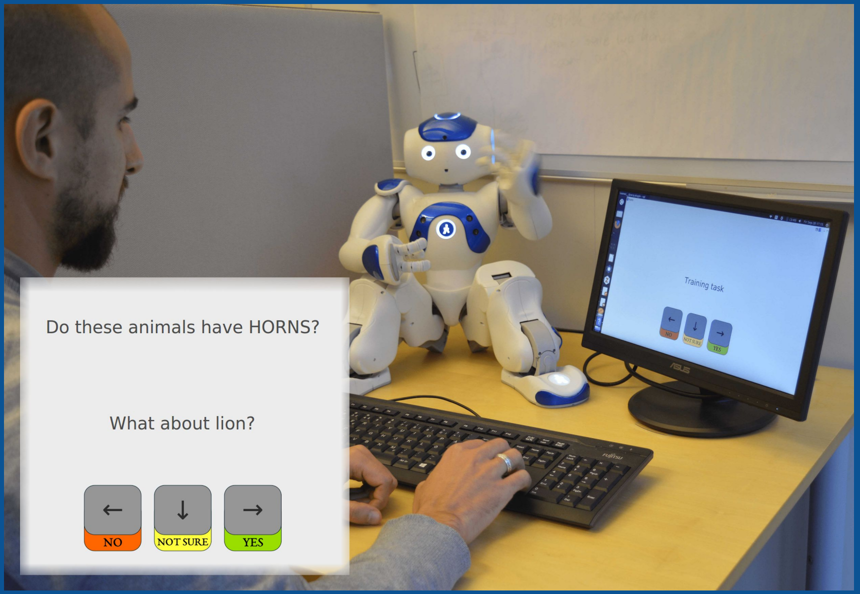


What about real users?

User study: 26 participants, the 3 strategies as conditions (within-subject).

Data logged:

- **NASA TLX**
- **Q&A, response times, prediction power**
- **Overall preferences**



What about real users?

User study: 26 participants, the 3 strategies as conditions (within-subject).

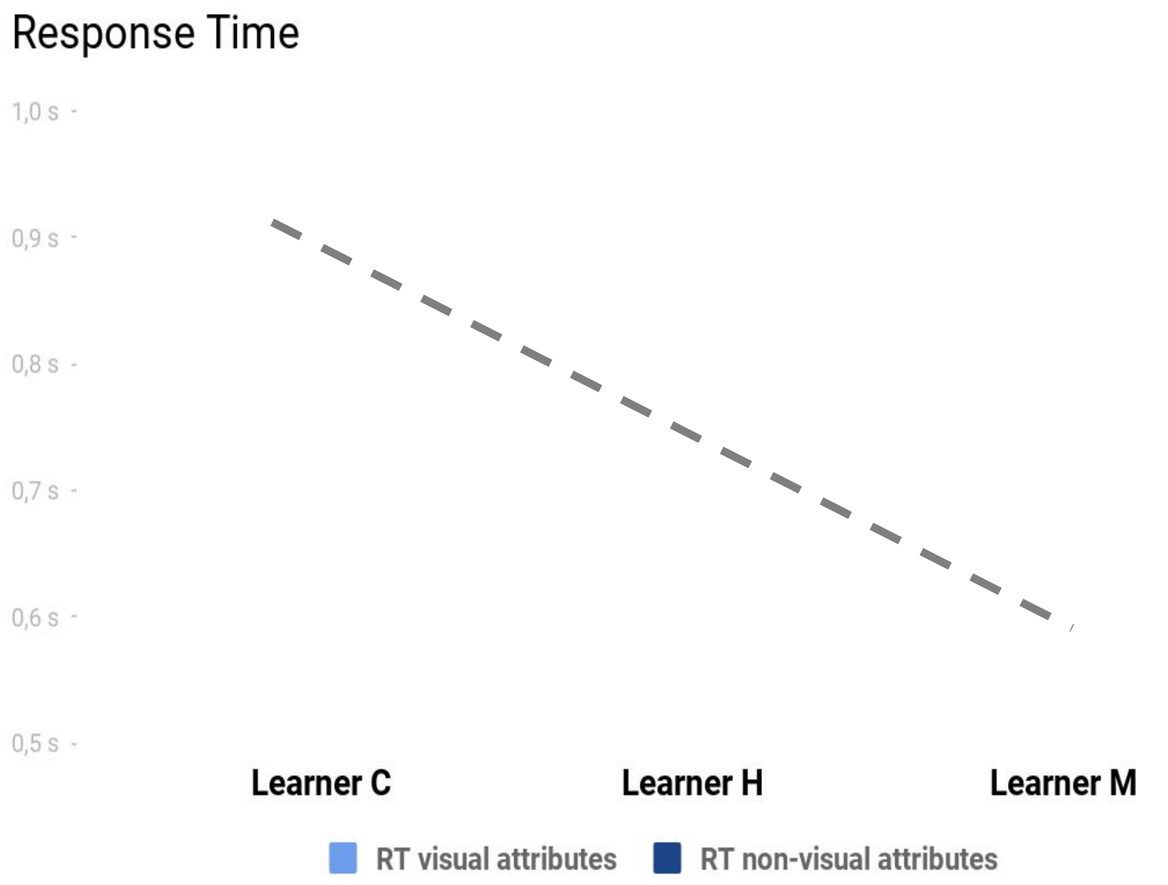
Data logged:

- **NASA TLX**
- **Q&A**, **response times**, **prediction power**
- **Overall preferences**

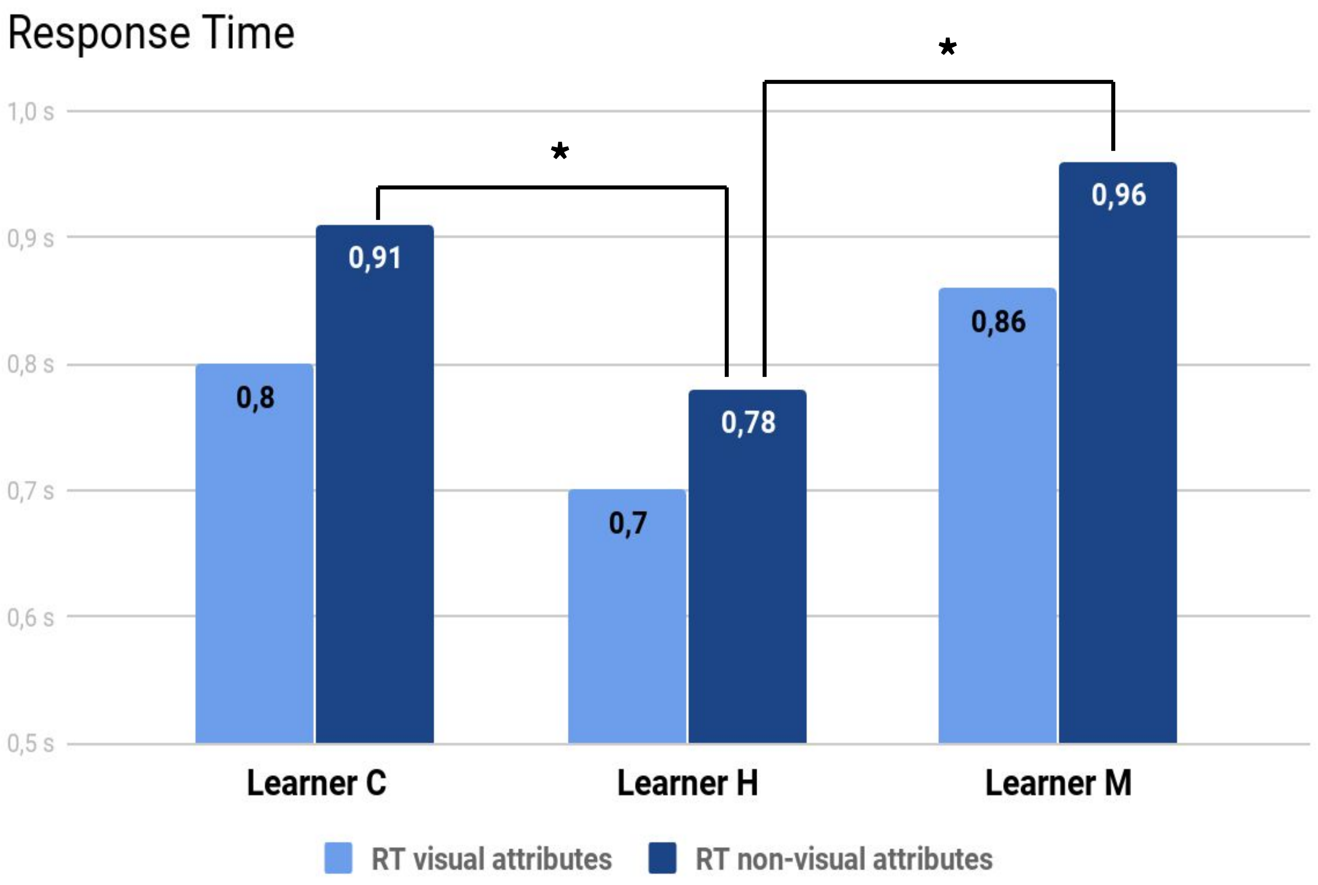
Our hypotheses:

Learner M makes the participants reply (a) **faster** and (b) with **less errors** compared to **Learner C**, with **Learner H** achieving **intermediate results**.

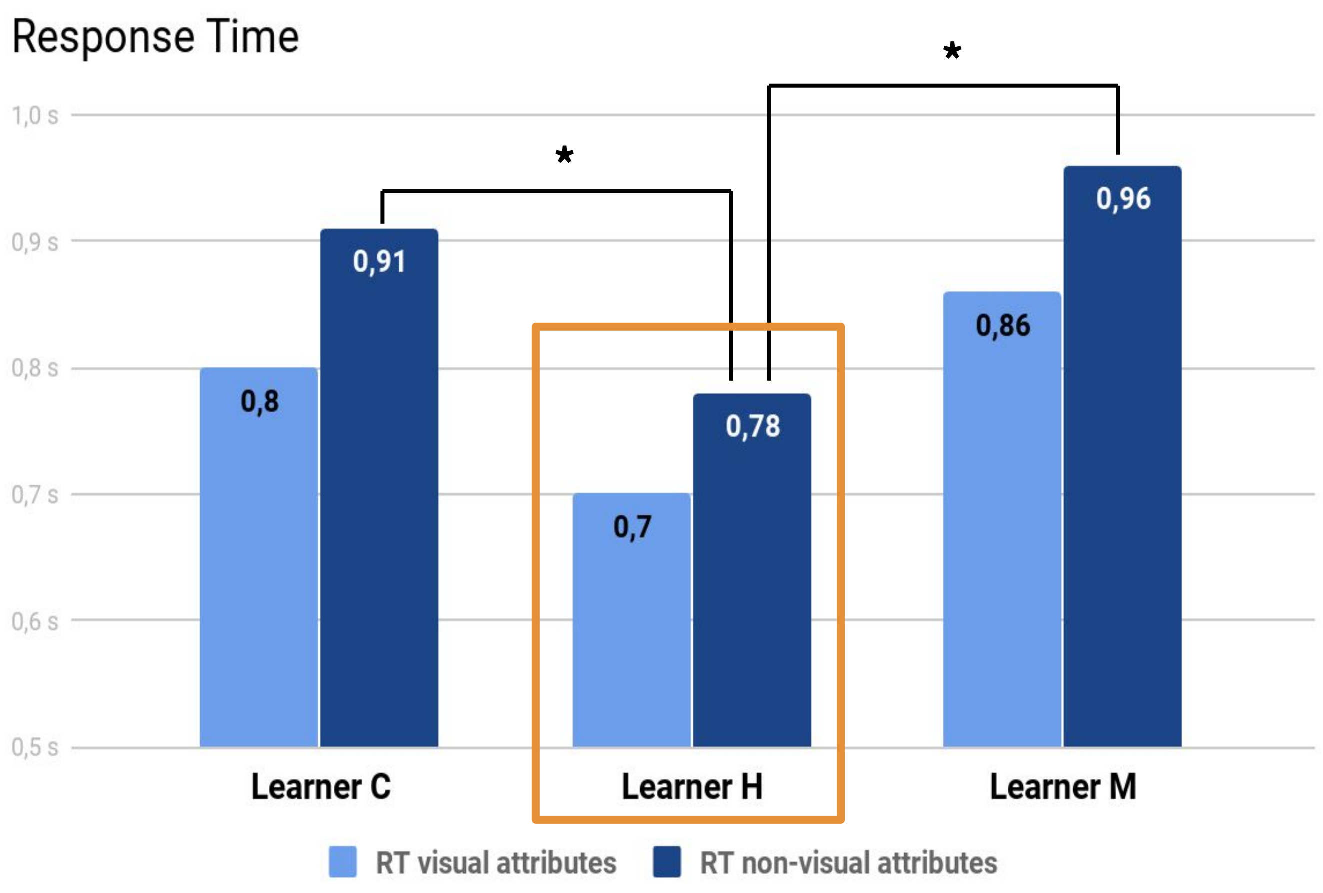
Results


Response Time
1,0 s
0,9 s
0,8 s
0,7 s
0,6 s
0,5 s
Learner C
Learner H
Learner M
RT visual attributes
RT non-visual attributes

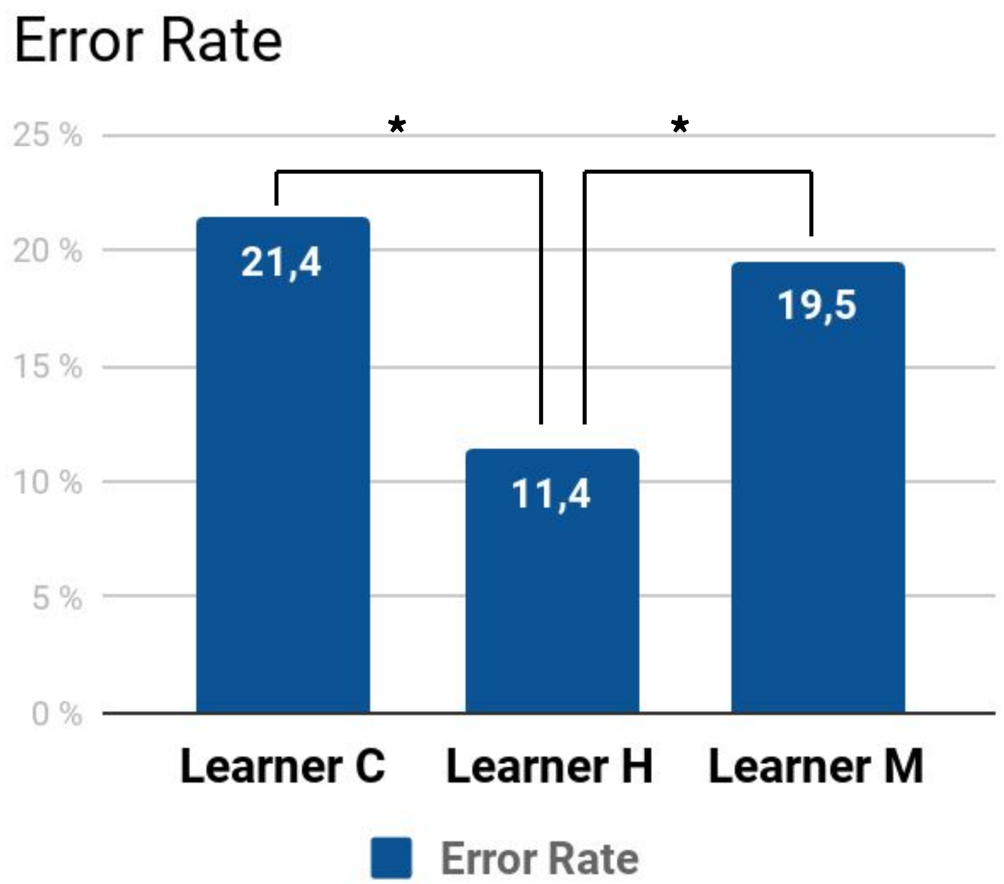
(Unexpected) Results



(Unexpected) Results



(Unexpected) Results



Discussion

- Higher response time and more errors for **Learner C**.

Discussion

- Higher response time and more errors for **Learner C**.
    - **stressful, unpredictable and requiring more thinking**

Discussion

- Higher response time and more errors for **Learner C**.
    - **stressful, unpredictable and requiring more thinking**
- Higher response time and more errors for **Learner M**.

Discussion

- Higher response time and more errors for **Learner C**.
    - **stressful, unpredictable and requiring more thinking**
- Higher response time and more errors for **Learner M**.
    - **easy, natural and predictable**

Discussion

- Higher response time and more errors for **Learner C**.
    - **stressful, unpredictable and requiring more thinking**
- Higher response time and more errors for **Learner M**.
    - **easy, natural and predictable**
    - **too easy? lowering attention or cause boredom**

Discussion

- Higher response time and more errors for **Learner C**.
  - **stressful, unpredictable and requiring more thinking**
- Higher response time and more errors for **Learner M**.
  - **easy, natural and predictable**
  - **too easy? lowering attention or cause boredom**
  - **too predictable? using the same (maybe wrong) answer**

Discussion

- Overall preferences:

Discussion

- Overall preferences:

- **Learner C as efficient → Mitigating difficulty!**

Discussion

- Overall preferences:

- **Learner C as efficient → Mitigating difficulty!**
- **Learner M as useless → Frustration and boredom!**

Discussion

- Overall preferences:

- **Learner C as efficient ⟶ Mitigating difficulty!**
- **Learner M as useless ⟶ Frustration and boredom!**
- **AVOID USELESS QUESTIONS!**

Conclusions

Can efficiency-driven Active Learning counter its own benefits?

Conclusions

Can efficiency-driven Active Learning counter its own benefits?

If we consider in the equation **non-oracle users**, **yes**!
But we just scratched the surface...

- We need a better understanding of interaction aspects that can affect learning
- Strategies that can **adapt to the specific user**

Teacher-Aware Active Robot Learning

Mattia Racca, Antti Oulasvirta and Ville Kyrki

mattia.racca@aalto.fi

Can efficiency-driven Active Learning counter its own benefits?

If we consider in the equation **non-oracle users** and the **interaction**, **yes**!

Thank you for the attention!

Code available at **github.com/MattiaRacca**

Aalto University
School of Electrical Engineering

Tree building algorithm

Algorithm 1 Build Entity-Category tree

Input: Entity set $\mathcal{E}$, Root of the tree $\mathcal{R}$, WordNet
Output: Entity-Category tree $\mathcal{T}$, Category set $\mathcal{C}$

Initialized tree $\mathcal{T}$ with root in $\mathcal{R}$
for all entities $e \in E$ **do**
    # find WordNet hypernym path leading from e *to* $\mathcal{R}$
    $\rho \leftarrow \{\}$; $\omega \leftarrow$ parent node of e
    while $\omega \neq \mathcal{R}$ **do**
        Append ω to ρ
        $\omega \leftarrow$ parent node of ω
    end while
    Add ρ to $\mathcal{T}$
end for
Prune trivial nodes from $\mathcal{T}$ (nodes with a single child)
$\mathcal{C} \leftarrow$ all non-terminal nodes of $\mathcal{T}$

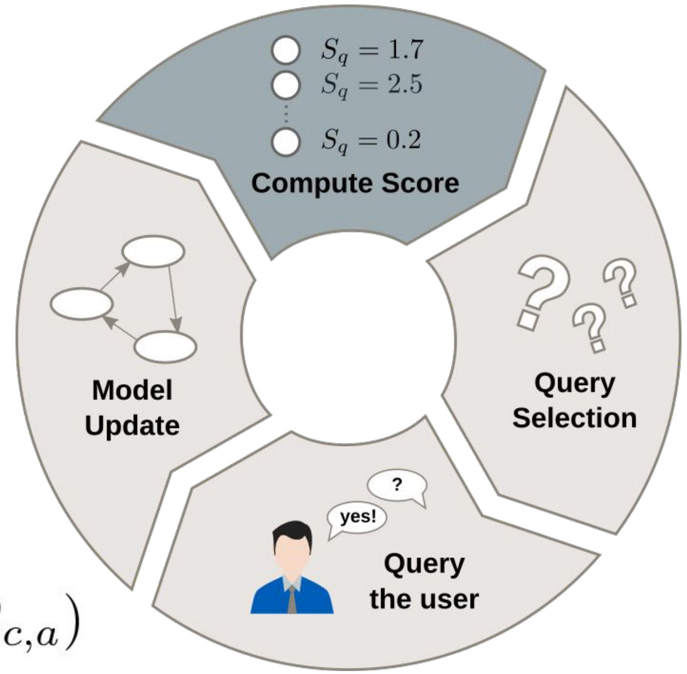
Attribute-Category Model

We model the probability of attribute *a* applying to category *c* as

$$p(a = x|c) \sim f_{c,a}(x|\theta_{c,a}),$$

and then we maintain a prior over these distribution. We can then compute the probability of *a* applying to entity *e* as

$$p(a = x|e) \sim f_{e,a}(x) = \sum_{c \in C} \bar{w}_{c,e} f_{c,a}(x|\theta_{c,a})$$



and therefore predict attribute entities pairs, given our current model. The update step of the model is the computation of the posterior distributions given the user answer *r* as an observation.

$$p(\theta_c|q_e, r) \propto p(\theta_c|\alpha_c, \beta_c)p(q_e, r|\theta_c) =$$

$$= \begin{cases} \text{Beta}(\theta_c|\alpha_c + \bar{w}_{c,e}, \beta_c) & \text{if } r = \text{yes} \\ \text{Beta}(\theta_c|\alpha_c, \beta_c + \bar{w}_{c,e}) & \text{if } r = \text{no} \end{cases}$$

Scores for each active learner

Learner C

$$s_{q,\mathbf{C}} = \mathbb{H}(\hat{f}_e(x)),$$

where $\hat{f}_e(x) = \mathrm{Ber}(x|\theta = \sum_c \bar{w}_{c,e}\theta_c)$ is an approximation of the full Bernoulli Mixture, as the entropy of a Bernoulli Mixture cannot be computed in close form.

Learner M

$$s_{q,\mathbf{M}} = \exp(-\delta d(e,p)),$$

where $d(e,p)$ is the distance (used in Eq. 3) between the entity e target of query q and the entity p target of the *previous query* and δ being a scale parameter similar to γ.

Learner H

$$s_{q,\mathbf{H}} = \phi s_{q,\mathbf{C}} + (1-\phi) s_{q,\mathbf{M}},$$

where parameter $\phi \in [0,1]$ controls the trade-off between the other two strategies.

Assumption choice