ResponseRank:
Data-Efficient Reward Modeling through **Preference Strength** Learning

Paper

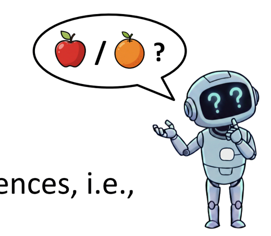
Contact

Timo Kaufmann[1,2], Yannick Metz[3], Daniel Keim[3], and Eyke Hüllermeier[1,2,4]

Background: Preference Learning for RLHF

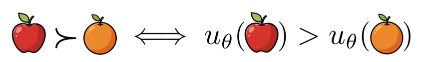
Reinforcement learning from human feedback (RLHF) is used to **fine-tune language models** and **train control policies** from comparative human feedback.

We aim to learn **better reward models** with **preference-strength-aware** learning using **auxiliary signals** with a monotone relationship to strength (e.g., response time).



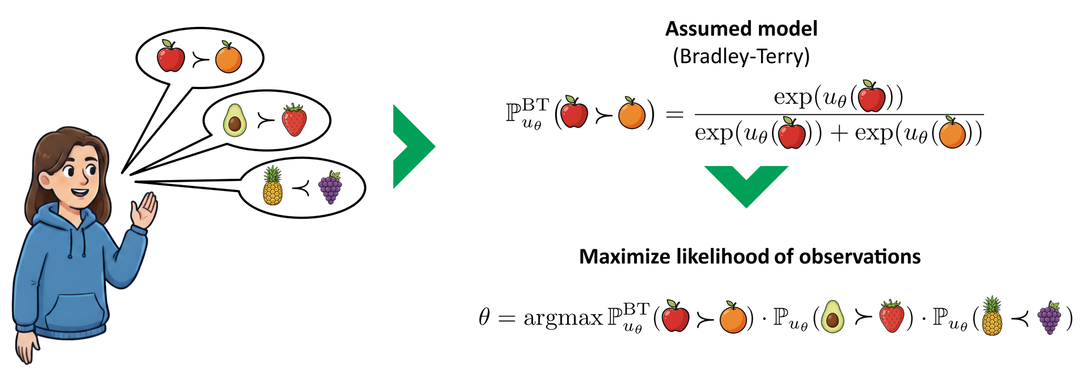
Here: Fruit preferences for illustration.

We want to learn utilities u_θ consistent with preferences, i.e.,



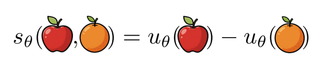
and additionally learn the **strength** of each preference.

The standard approach



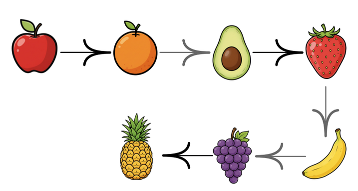
Strength under Bradley-Terry

We define **strength as utility difference:**



Strength is identifiable under BT, but only given unrealistic assumptions!

Assumption: A **fixed set of objects** with a **strongly connected preference graph.**

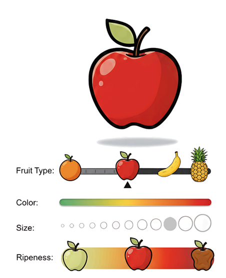


› The maximum likelihood estimate of the utilities is identifiable up to additive shift ($\mathbb{P}^{\mathrm{BT}}_{u}(\cdot) = \mathbb{P}^{\mathrm{BT}}_{u+c}(\cdot)$).

› Strengths are exactly identifiable

$s_u(a,b) = u(a) - u(b) = (u(a)+c) - (u(b)+c) = s_{u+c}(a,b)$

RLHF in practice: Utility functions over **parametric objects** with **isolated comparisons.**



› Only the order of some utilities is specified, **no information about strength.**

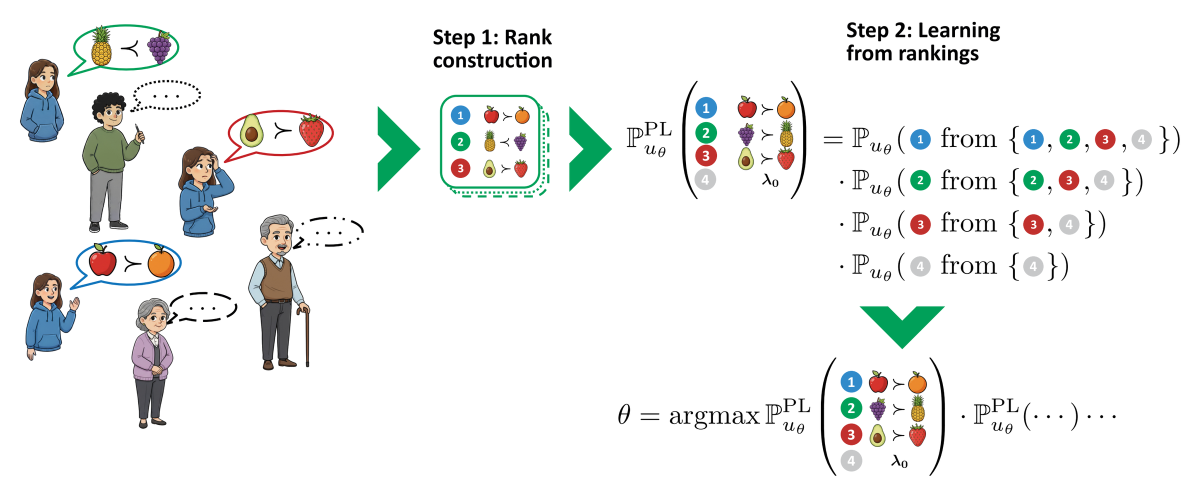
› Strength can only be learned through generalization between comparisons.

TL;DR

We learn **preference strength** from **implicit rankings** derived from locally valid relative signals.

› **Accurate** reward models, improved **data efficiency**, and **better policies**.

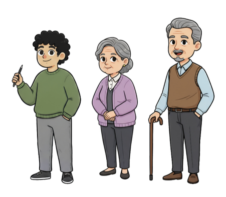
The ResponseRank Method



Step 1: Rank construction

Uses a strength signal (e.g., response time) with **few assumptions**:

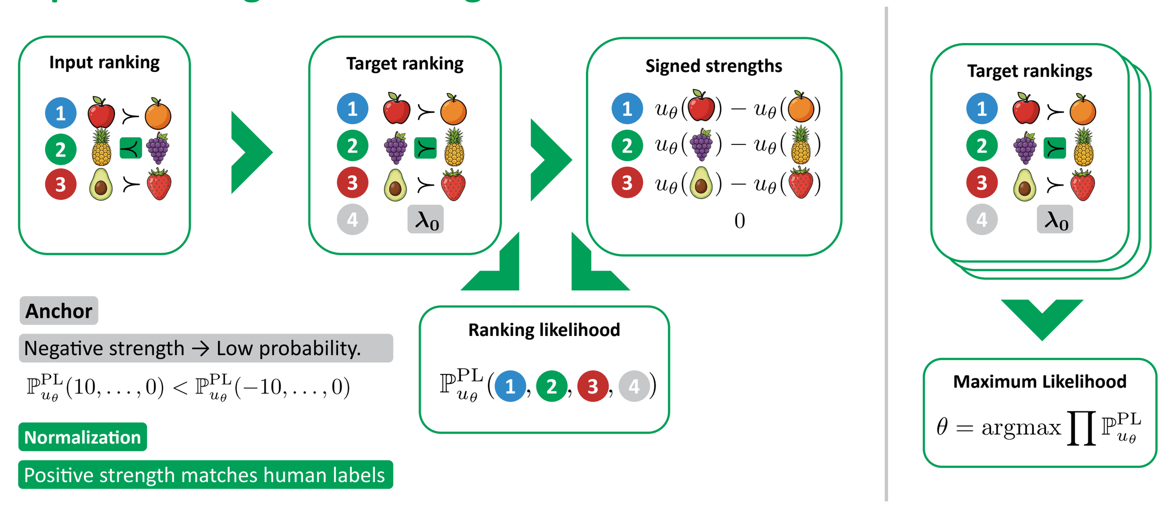
- **Local validity** within each ranking, enabling use of signals such as response time.
- A **monotone** relationship to strength (we need only ordinal information).



Beware of mixing response time rankings!

› Stratification

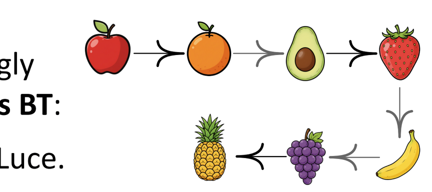
Step 2: Learning from rankings



Properties

Strength learning

Under idealized conditions (fixed set of objects, strongly connected preference graph), **ResponseRank matches BT**:

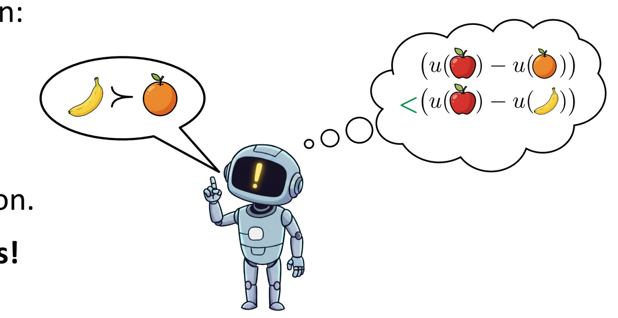


- Strengths are identifiable up to shift under Plackett-Luce.
- The antisymmetry of the pairwise strength ($s_{u_\theta}(a,b) = -s_{u_\theta}(b,a)$) prevents shifts.

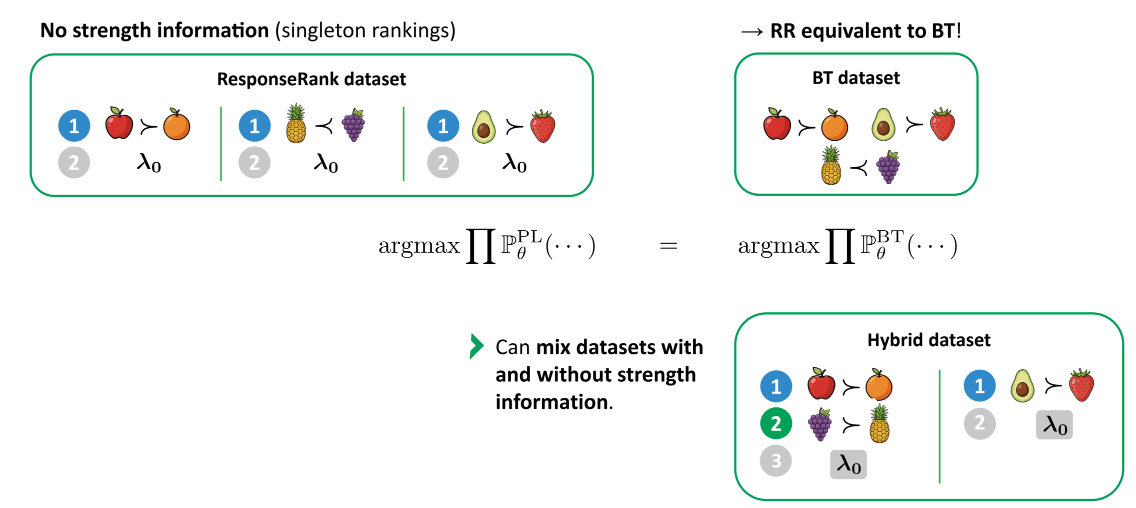
› **Strength is identifiable** › utilities up to shift.

With isolated comparisons of parametric objects, only ResponseRank has strength information:

- BT: No explicit strength information. Some strength through generalization.
- ResponseRank: Partial information about strength order. More through generalization.
- Sufficient for **inferring unseen preferences!**



Reduction to Bradley-Terry

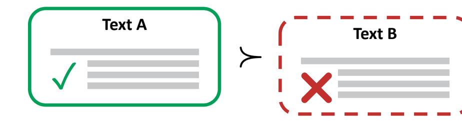


$$\operatorname{argmax} \prod \mathbb{P}^{\mathrm{PL}}_{\theta}(\cdots) = \operatorname{argmax} \prod \mathbb{P}^{\mathrm{BT}}_{\theta}(\cdots)$$

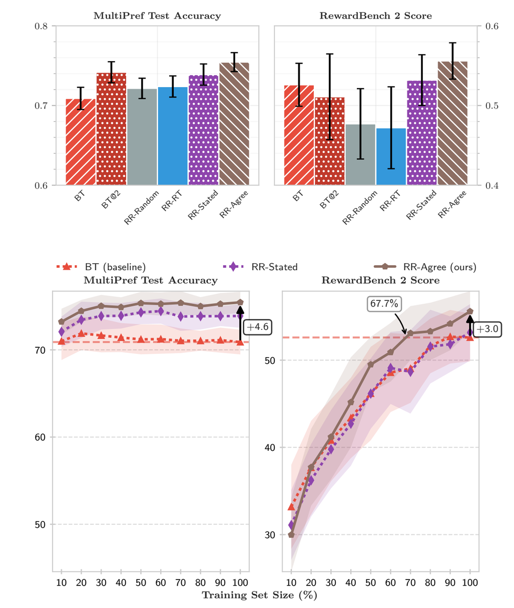
› Can **mix datasets with and without strength information.**

Empirical Evaluation

Reward models for LLMs



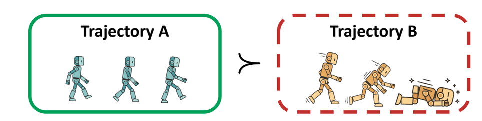
- We train reward models with BT and ResponseRank.
- We train on MultiPref, evaluate on RewardBench.
- Strength proxies
  - Response time
  - Stated strength (slight/clear)
  - Inter-annotator agreement $= \left(1.0 \cdot (n_c^+ - n_c^-) + 0.5 \cdot (n_s^+ - n_s^-)\right)/N$



Takeaway

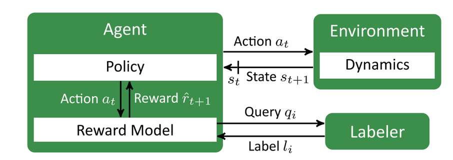
- Response time is not useful on this dataset.
- Agreement improves **accuracy** and **sample efficiency.**

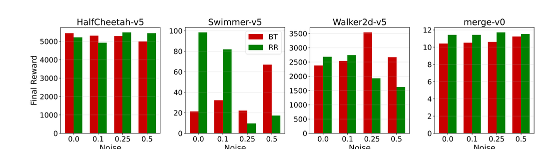
RL control



Environment (noise free)	Final reward (frac. of BT)
HalfCheetah-v5	5215.2 (96.0%)
Swimmer-v5	98.3 (**463.7%**)
Walker2d-v5	2679.7 (**113.2%**)
Merge-v0	11.4 (**109.6%**)

- RL control in simulation (MuJoCo, Highway merge)

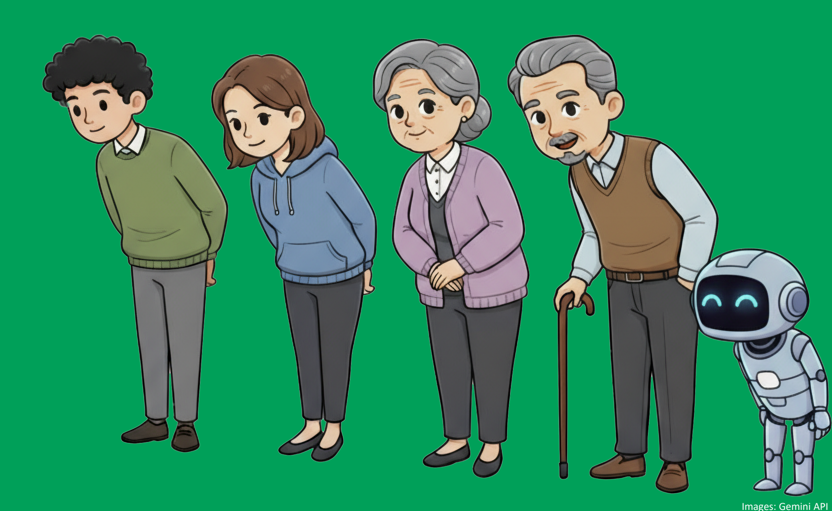




- Synthetic preferences (oracle rewards)
- Synthetic strength (reward difference)
- This is a proof of concept. More evaluation needed in this domain. Noise sensitivity is likely due to partition size.

Takeaway

Not only better reward models, but also better policies!



1) LMU Munich
2) Munich Center of Machin. Learning
3) University of Konstanz
4) German Research Center for Artificial Intelligence

NEURAL INFORMATION PROCESSING SYSTEMS

Timo Kaufmann Yannick Metz Daniel Keim Eyke Hüllermeier