模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

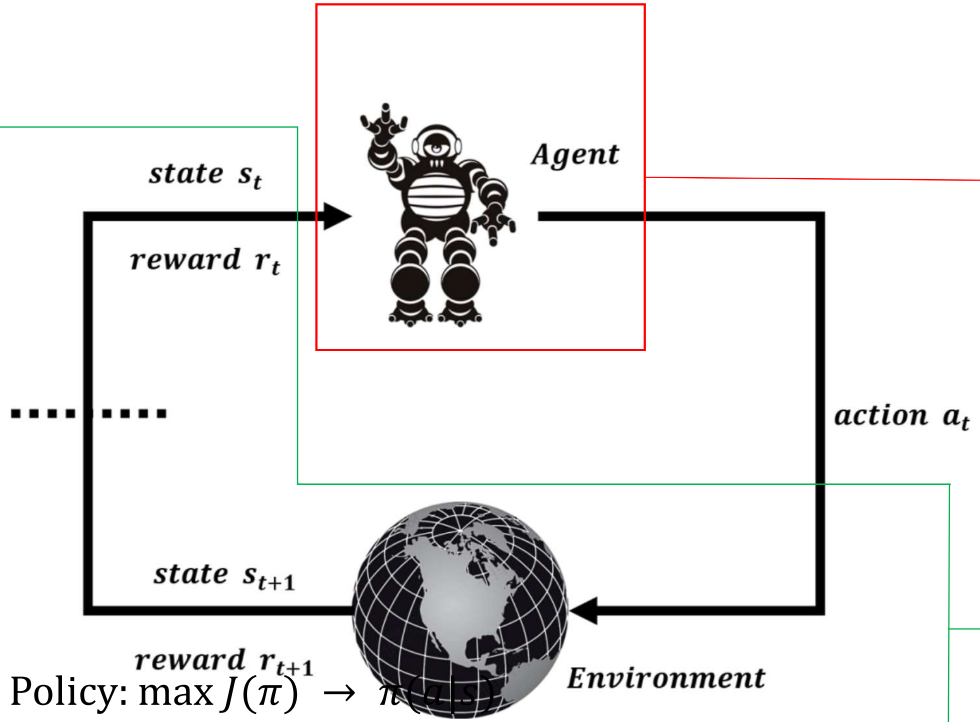
Boosting Offline Reinforcement Learning with Action Preference Query

Qisen Yang [*1] **Shenzhi Wang** [*1] **Matthieu Gaetan Lin** [2] **Shiji Song** [1] **Gao Huang** [1]

[*]Equal contribution [1]Department of Automation, BNRist, Tsinghua University, Beijing, China [2]Department of Computer Science, BNRist, Tsinghua University, Beijing, China. Correspondence to: Gao Huang <gaohuang@tsinghua.edu.cn>.

ICML2023

Background



Policy: $\max J(\pi) \rightarrow \pi(a|s)$

policy: $\pi(a|s)$

(actor)

value function

(critic)

- state value function: $V(s)$
- state-action value function: $Q(s,a)$

MDP=$(S, A, R, \gamma, P, \rho_0)$

S: state space　　A: action space

R: reward fuction　　γ: discount factor

P: state transition function　　$s_{t+1} \sim P(\cdot\,|s_t, a_t)$

ρ_0: distribution of initial state　$s_0 \sim \rho_0(s)$

Objective: $J(\pi) = \mathbb{E}_{s_0\sim\rho_0, a\sim\pi(\cdot|s), s'\sim P(\cdot|s,a)}[\sum_{t=0}^{T} \gamma^t r(s_t, a_t)]$

Background

$$V^{\pi}(s) = \mathbb{E}_{s_t,a_t\sim\rho_\pi}[\sum_{t=0}^{T}\gamma^t r(s_t,a_t)\,|s_0 = s]$$

$$Q^{\pi}(s,a) = \mathbb{E}_{s_t,a_t\sim\rho_\pi}[\sum_{t=0}^{T}\gamma^t r(s_t,a_t)\,|s_0 = s, a_0 = a]$$

$$\Rightarrow \begin{cases} V^{\pi}(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}[Q(s,a)] \\ Q^{\pi}(s,a) = r(s,a) + \gamma\mathbb{E}_{s'\sim P(\cdot|s,a)}V^{\pi}(s') \end{cases}$$

Bellman function

$$V^{\pi}(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}[r(s,a) + \gamma\mathbb{E}_{s'\sim P(\cdot|s,a)}V^{\pi}(s')]$$

$$Q^{\pi}(s,\mathrm{a}) = r(s,a) + \gamma\mathbb{E}_{s'\sim P(\cdot|s,a)}\mathbb{E}_{a'\sim\pi(\cdot|s')}[Q(s',a')]$$

policy iteration
- policy evaluation: use trajectories of π to evaluate Q and V
- policy improvement: use Q and V to update π e.g $\pi_{k+1}(a|s) = \mathrm{argmax}_a\, Q^{\pi_k}(s,a)$

Background

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing

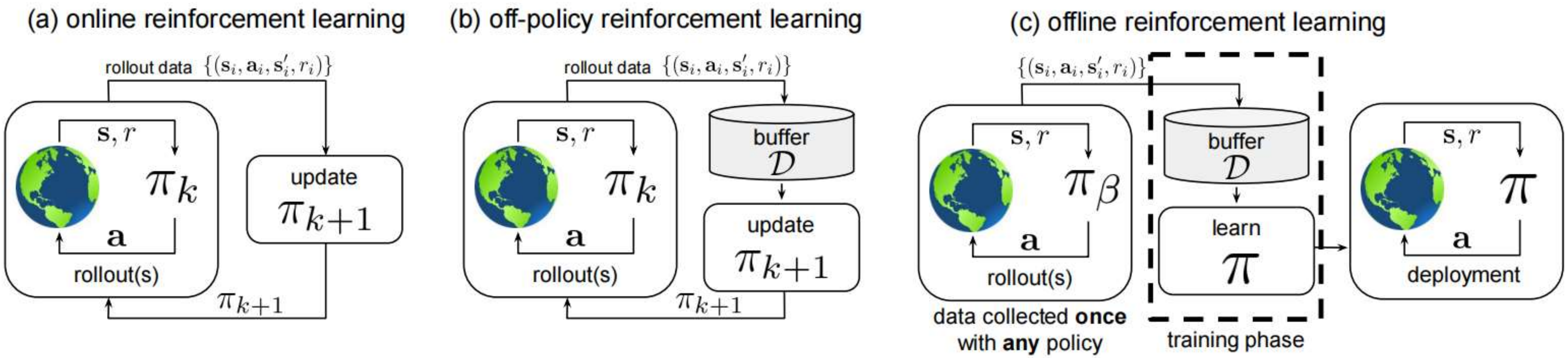
Disadvantage of RL : sample inefficiency

main reasons

- loss_V $=\text{MSE\_loss}(\mathbb{E}_{a\sim\pi(\cdot|s)}[r(s,a)+\gamma\mathbb{E}_{s'\sim P(\cdot|s,a)}V^{\pi}(s')],\ V^{\pi}(s))$
- The trade-off between exploration and exploitation

Because of the large amount of interaction with the environment, RL is not suitable for some high-risk environments, such as autonomous driving and healthcare.

Background



The main idea of offline RL is keeping pessimism, that is, learning a policy which is close to the behavior policy.

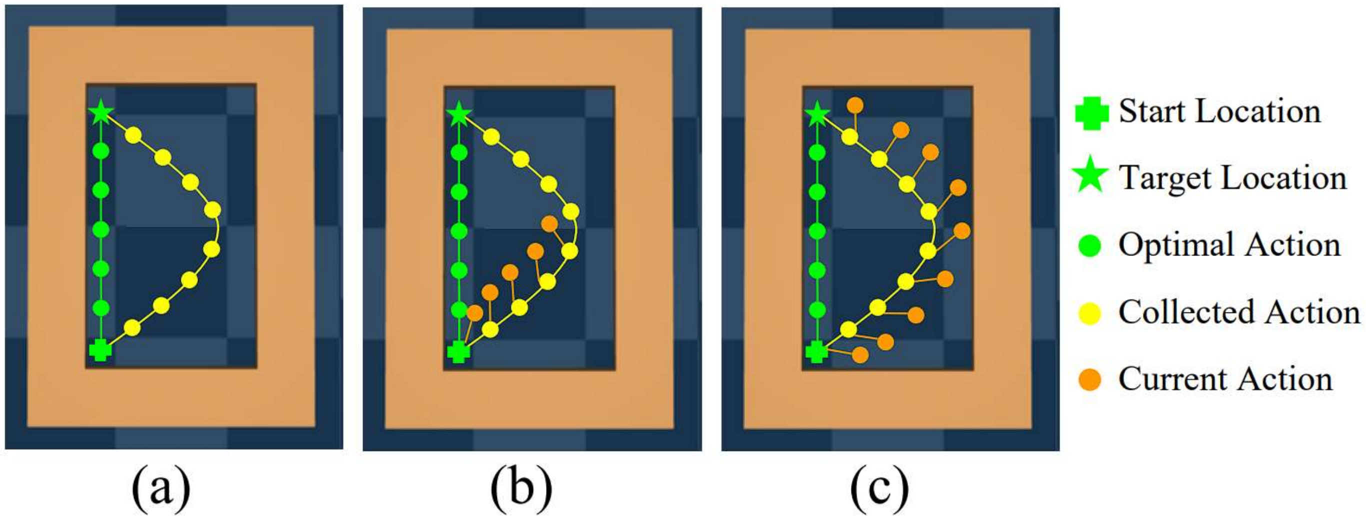
TD3+BC:

$$\pi = \underset{\pi}{\operatorname{argmax}} \mathbb{E}_{(s,a)\sim\mathcal{D}}[Q(s,\pi(s))] \longrightarrow \pi = \underset{\pi}{\operatorname{argmax}} \mathbb{E}_{(s,a)\sim\mathcal{D}}\left[\lambda Q(s,\pi(s)) - \boxed{(\pi(s)-a)^2}\right]$$

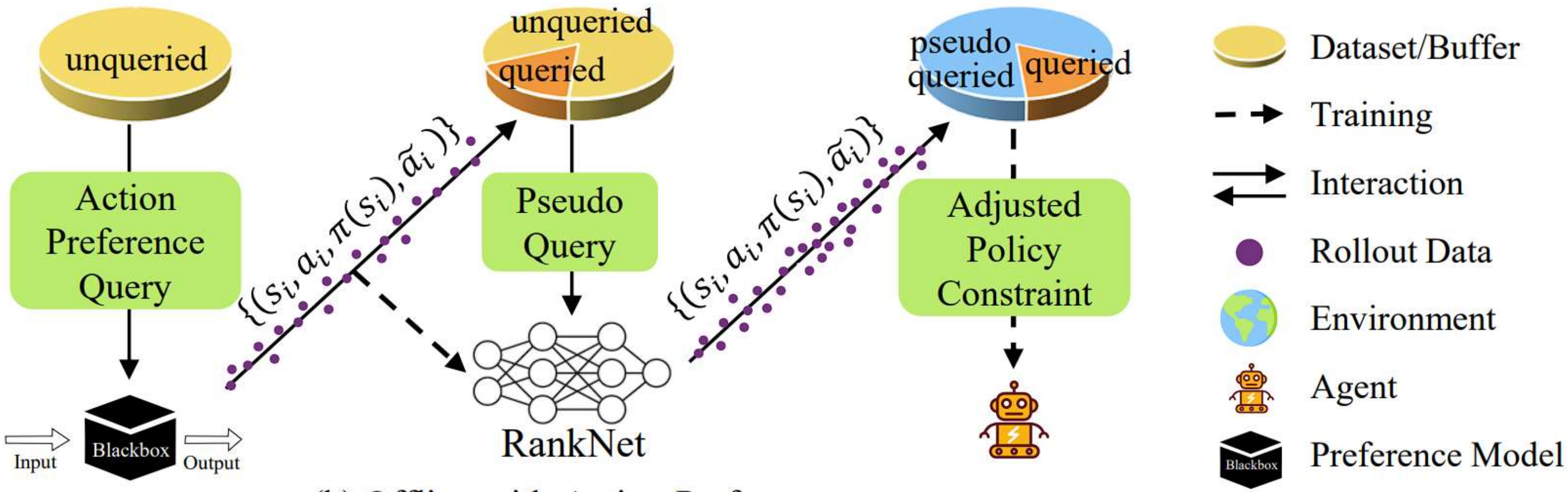
Motivation

Boosting Offline Reinforcement Learning with Action Preference Query

- Some actions in the dataset are of low quality, which could cause erroneous estimate problem.
- Action preference queries are easier to obtain in real-world environments.
- In some high-stake scenarios, limited online interactions can be inaccessible when online fine-tuning.



Method



(b) Offline-with-Action-Preferences

1. Action Preference Query

while training, query the preferred actions according to ranking criterion

ranking criterion $$l_i = (\pi(s_i) - a_i)^2$$

the preferred action $$\tilde{a}_i = G(s_i, a_i, \pi(s_i)) = \underset{a \in \{a_i, \pi(s_i)\}}{\arg\max}\ Q^*(s_i, a)$$

Method

2. Pseudo Query with RankNet

a query dataset $\mathcal{D}_q = \{(s_k, a_k, \pi^k(s_k), \tilde{a}_k) \mid k = 1, 2, \cdots, M\}$

ranking function f_r

RankNet $o_k = f_r(s_k, a_k) - f_r(s_k, \pi^k(s_k)))$ $\quad P_k = e^{o_k}/(1 + e^{o_k})$

$$C_k = C(o_k) = -\bar{P}_k \log P_k - (1 - \bar{P}_k)\log(1 - P_k)$$

pseudo query $G(s_i, a_i, \pi(s_i)) = \underset{a \in \{a_i, \pi(s_i)\}}{\arg\max} f_r(s_i, a)$

3. Adjusted Policy Constraint

$$\pi = \underset{\pi}{\operatorname{argmax}} \mathbb{E}_{(s,a)\sim\mathcal{D}} \left[\lambda Q(s, \pi(s)) - (\pi(s) - \boxed{a})^2\right] \Longrightarrow \pi = \underset{\pi}{\arg\max} \mathbb{E}_{(s,a)} \left[\lambda Q(s, \pi(s)) - (\pi(s) - \boxed{\tilde{a}})^2\right]$$

Proposition 3.1 (Perfect preference case)**.** *Consider the case with perfect preferences, i.e., $\forall(s,a)$, the state-action value function $Q^*(s,a)$ used for the action preference query is accurate. Then π_β and $\tilde{\pi}_\beta$ satisfy:*

$$\begin{aligned}&\eta(\tilde{\pi}_\beta)-\eta(\pi_\beta)\\&\quad\approx \mathbb{E}_{s\sim\mathcal{D}}\left[Q^*(s,\tilde{\pi}_\beta(s))-Q^*(s,\pi_\beta(s))\right]\geq 0.\end{aligned} \tag{7}$$

Proposition 3.2 (Imperfect preference case)**.** *Consider the case where preferences probably have errors. Denote the accurate state-action value function as $Q^*(s,a)$ and the faulty function as $\hat{Q}^*(s,a)$. Then $\forall \hat{Q}^*$ satisfying $D_{\mathrm{TV}}^{\tilde{\pi}_\beta}(\hat{Q}^*,Q^*)\leq\tilde{\alpha}, D_{\mathrm{TV}}^{\pi_\beta}(\hat{Q}^*,Q^*)\leq\alpha$, it holds that*

$$\begin{aligned}\eta(\tilde{\pi}_\beta)-\eta(\pi_\beta)\gtrsim \mathbb{E}_{s\sim\mathcal{D}}\Big[&\hat{Q}^*(s,\tilde{\pi}_\beta(s))\\&-\hat{Q}^*(s,\pi_\beta(s))\Big]-2(\tilde{\alpha}+\alpha)\overline{\rho}_{\pi_\beta},\end{aligned} \tag{8}$$

where $\overline{\rho}_{\pi_\beta}=\sup\{\rho_{\pi_\beta}(s), s\in\mathcal{S}\}\in\left[\frac{1}{|\mathcal{S}_\mathcal{D}|(1-\gamma)},\frac{1}{1-\gamma}\right]$ ($|\mathcal{S}_\mathcal{D}|$ denotes the number of different states in $\mathcal{D}$).

Algorithm 1 Offline-with-Action-Preferences

Require: Offline dataset $\mathcal{D}$, query dataset $\mathcal{D}_q$, training steps N_{train}, query intervals M_{inter}, query limit K_{total}.

Ensure: Policy π after optimization.

Initialize policy π and RankNet f_r.

Let the preferred actions $\tilde{a}_i = a_i, a_i\in\mathcal{D}$.

for $t=1\to N_{\text{train}}$ **do**

  Update the policy π by Equation (6).

  if $t \mod M_{\text{inter}}=0$ **then**

    Select $\frac{K_{\text{total}}M_{\text{inter}}}{N_{\text{train}}}$ samples from the offline dataset $\mathcal{D}$ according to the ranking criterion.

    Conduct action preference query by Equation (2).

    Add queried samples into the query dataset $\mathcal{D}_q$.

    for epoch = 0, 1, $\cdots$, until convergence **do**

      Train the RankNet f_r with the query dataset $\mathcal{D}_q$ by Equation (3).

    end for

    Conduct pseudo queries on the rest of the samples in the offline dataset $\mathcal{D}$ by Equation (4).

  end if

end for

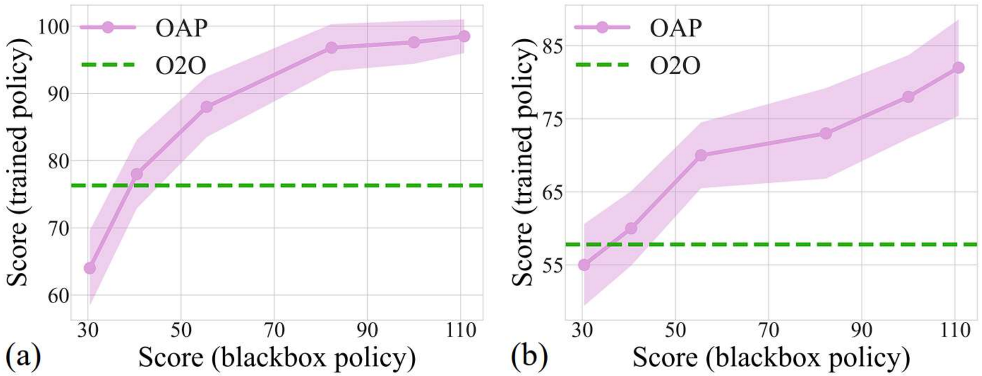
Experiment

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing

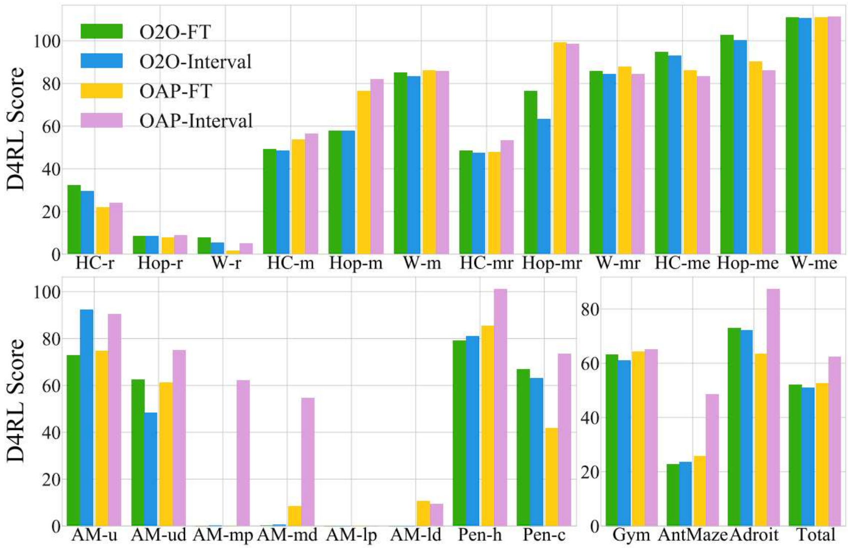
Dataset	Offline	Online	Online-Mix	Offline-to-Online	OAP
halfcheetah-random-v2	11.1 ± 1.3	19.2 ± 4.1	28.3 ± 2.1	32.3 ± 1.9	24.0 ± 1.6
hopper-random-v2	8.7 ± 1.6	14.5 ± 12.5	10.1 ± 0.4	8.6 ± 1.2	8.8 ± 1.8
walker2d-random-v2	1.8 ± 1.5	1.0 ± 2.5	2.2 ± 1.5	7.7 ± 8.0	5.1 ± 5.1
halfcheetah-medium-v2	48.1 ± 0.2	19.2 ± 4.1	48.1 ± 0.2	49.2 ± 0.2	56.4 ± 4.3
hopper-medium-v2	55.8 ± 1.2	14.5 ± 12.5	58.4 ± 1.7	57.8 ± 2.0	82.0 ± 6.6
walker2d-medium-v2	83.2 ± 0.5	1.0 ± 2.5	79.2 ± 9.9	85.1 ± 0.9	85.6 ± 1.2
halfcheetah-medium-replay-v2	44.9 ± 0.1	19.2 ± 4.1	46.0 ± 0.4	48.3 ± 0.3	53.4 ± 1.9
hopper-medium-replay-v2	57.2 ± 10.2	14.5 ± 12.5	48.4 ± 3.4	76.3 ± 13.3	98.5 ± 2.5
walker2d-medium-replay-v2	81.1 ± 2.8	1.0 ± 2.5	76.7 ± 14.1	85.6 ± 1.7	84.3 ± 2.7
halfcheetah-medium-expert-v2	85.4 ± 3.3	19.2 ± 4.1	82.2 ± 5.2	94.5 ± 0.6	83.4 ± 5.3
hopper-medium-expert-v2	88.7 ± 2.9	14.5 ± 12.5	97.0 ± 7.6	102.5 ± 4.5	85.9 ± 6.6
walker2d-medium-expert-v2	110.9 ± 0.6	1.0 ± 2.5	110.1 ± 0.8	110.8 ± 0.4	111.1 ± 0.6
Gym Average	**56.4** ± 2.2	**11.6** ± 6.4	**57.2** ± 3.9	**63.2** ± 2.9	**64.9** ± 3.3
antmaze-umaze-v0	94.4 ± 2.7	0	0	72.8 ± 36.8	90.4 ± 5.2
antmaze-umaze-diverse-v0	51.0 ± 16.8	0	0	62.5 ± 31.2	75.0 ± 19.0
antmaze-medium-play-v0	1.4 ±0.8	0	0	0	62.0 ± 10
antmaze-medium-diverse-v0	1.0 ± 1.9	0	0	0.3 ± 0.4	54.5 ± 23.3
antmaze-large-play-v0	0	0	0	0	0
antmaze-large-diverse-v0	0	0	0	0	9.4 ± 8.4
AntMaze Average	**24.6** ± 3.7	**0**	**0**	**22.6** ± 11.4	**48.6** ± 11.0
pen-human-v1	84.8 ± 11.2	4.3 ± 7.4	8.5 ± 10.1	79.1 ± 14.5	101.2 ± 11.5
pen-cloned-v1	56.2 ± 16.3	4.3 ± 7.4	57.2 ± 29.5	66.8 ± 11.1	73.5 ± 13.0
Adroit Average	**70.5** ± 13.7	**4.3** ± 7.4	**32.8** ± 19.8	**72.9** ± 12.8	**87.4** ± 12.2
Average	**48.3** ± 3.8	**7.4** ± 4.6	**37.6** ± 4.3	**52.0** ± 6.4	**62.2** ± 6.5

Experiment

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing



work with faulty annotations



what if queries were focused instead of spaced

Dataset	OAP	OAP (inf)	OAP (w/ RN)
HC-r	24.0 ± 1.6	22.0 ± 1.4	11.0 ± 1.1
Hop-r	8.8 ± 1.8	13.2 ± 7.4	8.1 ± 0.8
W-r	5.1 ± 5.1	2.3 ± 2.0	1.9 ± 0.9
HC-m	56.4 ± 4.3	59.2 ± 1.5	48.2 ± 0.3
Hop-m	82.0 ± 6.6	92.8 ± 4.1	45.9 ± 1.5
W-m	85.6 ± 1.2	86.6 ± 0.3	84.8 ± 2.4
HC-mr	53.4 ± 1.9	51.3 ± 0.6	28.4 ± 3.1
Hop-mr	98.5 ± 2.5	101.9 ± 2.0	31.6 ± 3.5
W-mr	84.3 ± 2.7	84.9 ± 9.9	74.8 ± 5.5
HC-me	83.4 ± 5.3	84.1 ± 3.2	84.3 ± 5.3
Hop-me	85.9 ± 6.6	92.2 ± 8.2	80.6 ± 2.6
W-me	111.1 ± 0.6	109.4 ± 1.5	111.0 ± 0.4
Avg.	$\mathbf{64.9} \pm 3.3$	$\mathbf{66.7} \pm 3.5$	$\mathbf{50.9} \pm 2.3$

necessity for RankNet

Thanks