$Q^\star$ meets Thompson Sampling: Scaling up Exploration via HyperAgent

With application in Human-AI Alignment and Collaboration

Yingru Li

`https://richardli.xyz/`

June 25, 2024

Outline

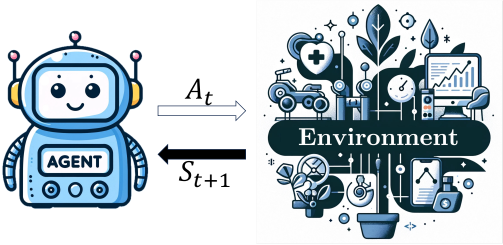
Motivation: RL under Resource Constraints

Existing solution and their limitations
- Our Contributions

HyperAgent for Efficient RL [LXHL24] **(ICML)**
- Theoretical insights with tabular representation
- Empirical evidence on efficient deep exploration with deep RL
- Reduce **sequential posterior approx.** to sequential random projection

GPT-HyperAgent for Continual Content Moderation

Reinforcement Learning Problem



Agent-Environment Interface.

- Interactive Experience:

$$A_0, S_1, A_1, S_2, \ldots, A_t, S_{t+1}, \ldots$$

Environment $M = (\mathcal{S}, \mathcal{A}, P)$

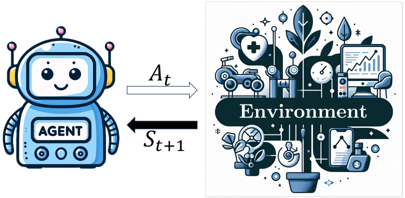
- State $S_{t+1} \sim P\left(\cdot \mid S_t, A_t\right)$ for $t = 0, 1, \ldots$.

Agent$(\mathcal{S}, \mathcal{A}, r, \mathcal{D}_t) \to \pi_t$ max long-term rewards

- **Reward** $R_{t+1} = r\left(S_t, A_t, S_{t+1}\right)$ preference
- **Data** $\mathcal{D}_t = \mathcal{D}_{t-1} \cup \left\{A_{t-1}, S_t\right\}$ **accumulated**.
- **Policy** $\pi_t = \text{Agent}(\mathcal{S}, \mathcal{A}, r, \mathcal{D}_t)$.
- **Action** $A_t \sim \pi_t(\cdot \mid S_t)$;
- **Objective** $\pi_{\text{agent}} = (\pi_0, \pi_1, \ldots)$ to maximize

$$\mathbb{E}\Big[\sum_{t=0}^{T-1} R_{t+1} \mid \pi_{\text{agent}}, M\Big]. \tag{1}$$

Challenges for Deploying RL in Real-world



Agent-Environment Interface.

- Interactive Experience:

$$\underbrace{A_0, S_1, A_1, S_2, \ldots, A_t, S_{t+1}, \ldots}_{\mathcal{D}}$$

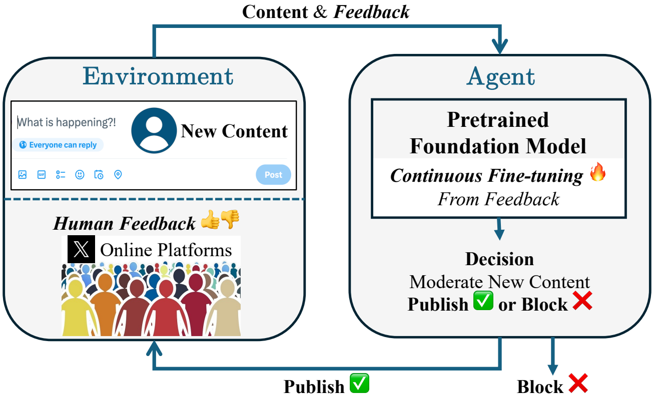
Complex Environment:

- $|\mathcal{S}| \approx 10^{100}$ **Large state space**
  - language, vision & audios, etc.
- $|\mathcal{D}| \uparrow$ **Data accumulates** as interacting.

Resource Constraints for Agent:

- **Bounded** Per-step Computation & Memory
  - e.g., **Real-time** decision-making in **online** recommendation **systems** [ZVR23].
- **Limited** Data Collection Budgets
  - e.g., **Human feedback** in LLMs [DAHVR24].
  - e.g., Scientific **experimental design**

Distributed Continual Content Moderation: Human-AI Collaboration



Content Moderation:

- **Safe Alignment: Filter** out **harmful** content according to human value.
- **Challenges** 1: Natural language input, etc $\rightarrow$ **Large** $|\mathcal{S}|$
- **Challenges** 2: **Real-time Safe Critical** decision-making
  - Adapting fast from limited human feedback while keeping track of the ongoing data stream $|\mathcal{D}|$.
- **Challenges** 3: **Partial** feedback
  - Only published content is available for feedback from crowd-sourced systems.

Outline

Motivation: RL under Resource Constraints

Existing solution and their limitations

- Our Contributions

HyperAgent for Efficient RL [LXHL24] **(ICML)**

- Theoretical insights with tabular representation
- Empirical evidence on efficient deep exploration with deep RL
- Reduce **sequential posterior approx.** to sequential random projection

GPT-HyperAgent for Continual Content Moderation

Development of RL Algorithms: A History of "Scale up!"

50s-60s
R. Bellman,
R. A. Howard,
D. Blackwell
Dynamic Programming
(Markov) Decision Processes

70s-90s
R.S. Sutton & A. Barto
C. J.C.H. Watkins
D. Bertsekas & J. Tsitsiklis
Temporal Difference, Q-Learning
Approximate (Neuro) DPs
Parallel and Distributed Computation

2010s --
DeepMind
OpenAI
Etc.
Deep RL (Advances in Algorithms and Computations)
AlphaGo & GPTs (Foundation Models)

Scale up!

Computation First

- **Per-step** $O(\text{poly}(|\mathcal{S}|))$ $\Rightarrow$ **Function Approximation** (FA), e.g. Neural Networks; (70s – 90s)
- **Per-step** $O(\text{poly}(|\mathcal{D}|))$ $\Rightarrow$ **Incremental Update** with SGD, Replay Buffer and/or Target Network.
- **FA + Incremental Update** $\Rightarrow$ **Bounded** $\tilde{O}(1)$ **Per-step Complexity** $\Rightarrow$ **Scalable** algorithm. (10s –)

Practical Advancements for "Efficient" Deep RL

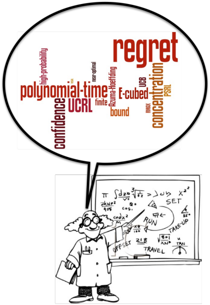
Algorithm	Components
DDQN (16)	Incremental SGD with experience replay (finite buffer) and target network
Rainbow (18)	(**DDQN**) + Prioritized replay, Dueling networks, Distributional RL, Noisy Nets.
BBF (23)	(**DDQN**) + Prioritized replay, Dueling networks, Distributional RL, Self-Prediction, Harder resets, Larger network, Annealing hyper-parameters.

Table: Components in **STOA algorithms**, e.g. DDQN [VHGS16], Rainbow [HMVH+18], BBF [SCC+23].

- ✓ **Scalable**: e.g. DDQN use incremental SGD with experience replay and target network.
- ✗ **Deployment inefficient**: **Complicated** components and **many** heuristics. **Hard to tune**.
- ✗ **Data inefficient**: e.g. BBF use ϵ-greedy **exploration strategy** which suffer **linear regret** in some environment, **provably** [Kak03, Str07, OVRRW19, DMM+22]. **In practice**, deep RL **data hungury** .

Principled Approaches for Data Efficiency

Goal: **Sequential decision-making** under **uncertainty** with **sublinear regret**.



R.A. Fisher, William R. Thompson (30s)
T. L. Lai and H. Robbins (80s)
Sequential design and allocations (MAB)

Before 90s

E3/PSRL/UCRL ...
Tabular RL

2000 --

Eluder dimension/Information Ratio
Bellman Rank/Bilinear Structure
Decoupling coefficient/DEC/GEC ...
Structural assumption and Algorithms for RL with function approximation

2014 --

Data First

Data Efficient Exploration

Posterior Sampling Reinforcement Learning (PSRL): **data-efficient exploration** strategy

- **Require**: Prior distribution $\mathbb{P}(M \in \cdot)$ for underlying model M.
- **For each episode** ℓ, denote t_ℓ the beginning time step
  - Sample $\hat{M}_\ell \sim \mathbb{P}\left(M \in \cdot \mid \mathcal{D}_{t_\ell}\right)$.
  - **Return** the optimal policy $\pi_\ell = \pi^{\hat{M}_\ell}$ under $\hat{M}_\ell$.

- Require **conjugacy** for **tractable posterior update** **(uncertainty estimation)**.
- **Only feasible** in simple environments:
  - Tabular MDP with dirichlet prior [Str00, OVR17] $\tilde{O}(H^2\sqrt{SAK})$ regret **sublinear in** K **episodes**.
  - Linear-Gaussian bandit [RVR16, RVRK$^+$18] $O(d\sqrt{T\log A})$ regret **subliear in** T **time steps.**

Data Efficient Exploration under Function Approximation (FA)

✗ Intractable Computation in **Posterior Sampling**:

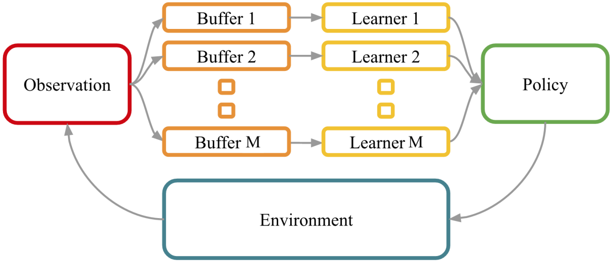
- **Model-based**: **No conjugacy** for exact Bayesian inference for posterior over transition models.
  - **[LL24] (AISTATS)**: **First prior-dependent** bound under FA and **improved prior-free** bound in the context of **linear mixture MDPs**.
- **Model-free**: Beyond conjugacy, sample from **intricate distribution over value functions** [Zha22, DMZZ21, ZXZ+22]

✗ Unbounded Per-step Complexity $\mathrm{poly}(|\mathcal{D}|)$ in **Approximate Posterior Sampling**:

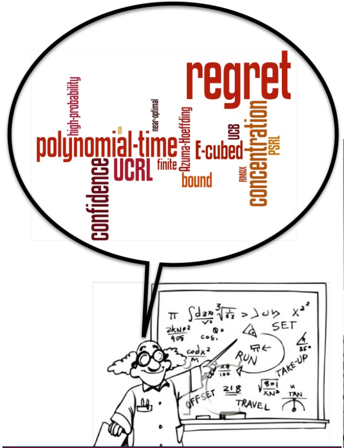
- **Store entire history and retrain for each episode**, e.g. RLSVI [OVRRW19], LSVI-PHE [ICN+21].
- **Langevin Monte-Carlo (LMC)** based methods [XZM+22, ILX+24]

- **Same issues for OFU**: (1) ✗ **Intractability** [JKA+17, JLM21, DKL+21, FKQR21, LLX+23]; (2) ✗ **Unbounded resource demands** as data accumulates [WSY20, AJZ23].

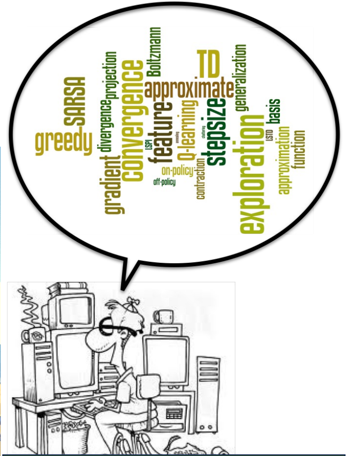
Ensemble Sampling for Approximate Posterior Sampling



- **Ensemble Sampling (ES)**: approximate the posterior distribution by **uniformly sampling from a set of ensemble models**. E.g., BootstrapDQN [OBPVR16], **Ensemble+** [OAC18, OVRRW19].
- ✓ Each ensemble perform incremental update, **no retraining**.
- ✗ **Computationally expensive** in practice: say, update > 100 neural networks for each time step.
- ✗ **No rigorous** understanding in terms of statistical and computational complexity.



Theory! Data First



Scale up! Computation First

Outline

Motivation: RL under Resource Constraints

Existing solution and their limitations

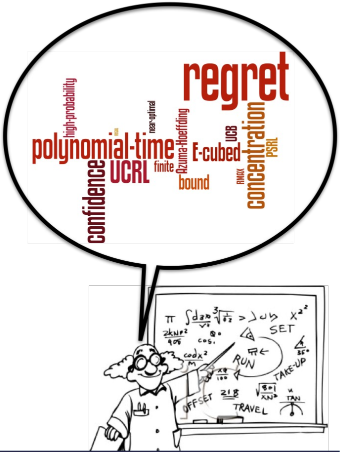
- Our Contributions

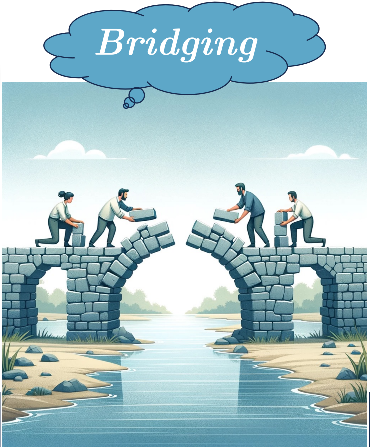
HyperAgent for Efficient RL [LXHL24] **(ICML)**

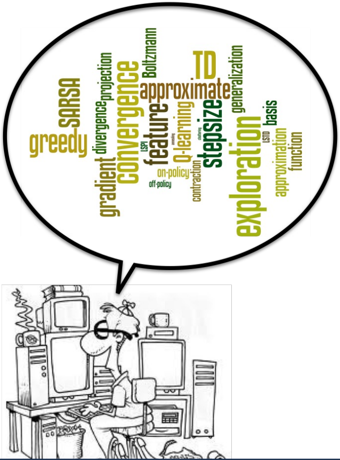
- Theoretical insights with tabular representation
- Empirical evidence on efficient deep exploration with deep RL
- Reduce **sequential posterior approx.** to sequential random projection

GPT-HyperAgent for Continual Content Moderation

Our HyperAgent [LXHL24] aims to ...







Theory! Data First

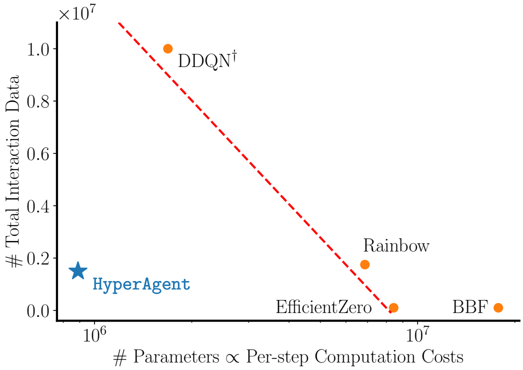
Scale up! Computation First

Preview of Contributions - For Practitioners

Algorithm	Components
DDQN (16)	Incremental SGD with experience replay (finite buffer) and target network
Rainbow (18)	(**DDQN**) + Prioritized replay, Dueling networks, Distributional RL, Noisy Nets.
BBF (23)	(**DDQN**) + Prioritized replay, Dueling networks, Distributional RL, Self-Prediction, Harder resets, Larger network, Annealing hyper-parameters.
HyperAgent	(**DDQN**) + **Hypermodel**

Table: Techniques used in different algorithms, e.g. DDQN [VHGS16], Rainbow [HMVH$^+$18], BBF [SCC$^+$23] and **our HyperAgent**.

- ✓ **Simple**: **Only one** additional component, **hypermodel**, compatiable with all feedforward DNN.
  $\Rightarrow$ **Easy to deploy** empirically.
- ✓ **Scalable**: **Incremental** SGD under DNN function approximation, same as DDQN;
  $\Rightarrow$ **bounded per-step computation**.



How much data and parameters to achieve **Human-level performance (1 IQM)** in Atari suite?

- ✓ **Data efficient**: only 15% data consumption of DDQN[VHGS16] by **DeepMind**. (1.5**M interactions**)
- ✓ **Computation efficient**: only 5% model parameters of BBF[SCC$^+$23] by **DeepMind**.
- **Ensemble+** [OAC18, OVRRW19] achieves a **mere** 0.22 **IQM score under** 1.5**M** interactions but necessitates **double the parameters of** `HyperAgent`.

Preview of Contributions - For Theoretists

	Practice in Deep RL			**Theory** in Tabular RL	
Algorithm	**Tractable**	**Incremental**	**Efficient**	**Regret**	**Per-step Computation**
PSRL	✗	✗	✗	$\tilde{O}(H^2\sqrt{SAK})$	$O(S^2A)$
RLSVI	✓	✗	✗	$\tilde{O}(H^2\sqrt{SAK})$	$O(S^2A)$
Ensemble+	✓	✓	●	N/A	N/A
`HyperAgent`	✓	✓	✓	$\tilde{O}(H^2\sqrt{SAK})$	$\tilde{O}(\log(K)SA + S^2A)$

- **HyperAgent** not only demonstrates superior empirical performance in deep RL benchmarks
- but also achieves **theoretical milestones**, i.e., the first method to achieve $\tilde{O}(\log K)$ **per-step computation** & **near-optimal regret** in **tabular** K**-episodic RL** among practically scalable algorithms.

Preview of Contributions - For Practitioners & Theoretists

Algorithmic Mechanism

- Value-based **approximate posterior sampling** via **hypermodel** and **index sampling** schemes.
- $\Rightarrow$ **Near-optimal regret bound** $\Rightarrow$ **Data-efficient** Exploration

$\Uparrow$

Key Lemma

- **Incremental approximation of posteriors** over value function **without conjugacy**.
- $\Rightarrow$ **Logarithmic** per-step **computation complexity** $\Rightarrow$ **Scalable** Uncertainty Estimation.

$\Uparrow$

- **Fundamental Tools** for **dynamic (non-i.i.d.) data**: **First Probability Tool** for **Sequential Random Projection** – a **non-trivial martingale extension** of Johnson-Lindenstrauss (JL). [Li24a]
- **Fundamental Tools** for **static data**: **Simple, Unified JL analysis** that covers existing and new JL construction that **traditional analysis cannot handle**. [Li24b]

Outline

Motivation: RL under Resource Constraints

Existing solution and their limitations

- Our Contributions

HyperAgent for Efficient RL [LXHL24] **(ICML)**

- Theoretical insights with tabular representation
- Empirical evidence on efficient deep exploration with deep RL
- Reduce **sequential posterior approx.** to sequential random projection

GPT-HyperAgent for Continual Content Moderation

HyperAgent: Introducing Hypermodel

- **Hypermodel**: $(\underbrace{f_\theta}_{\text{Paramatric function}}, \underbrace{P_\xi}_{\text{Reference distribution}})$ s.t.

  Index Sampling: $f_\theta(x, \underbrace{\xi}_{\text{Index sample } \xi \sim P_\xi})$ is an (approximate) **posterior predictive sample** on data x.

Example: predictive sampling from Linear-Gaussian model

- Suppose $\theta^* \sim N(\mu, \boldsymbol{\Sigma})$ where $\boldsymbol{\Sigma}$ represent the **model uncertainty**.
- **Box-Muller Transform**: $P_\xi = N(0, I_M)$, $\theta = (\boldsymbol{A} \in \mathbb{R}^{d \times M}, \mu \in \mathbb{R}^d)$

$$\xi \sim P_\xi \Rightarrow f_\theta(x, \xi) := \langle x, \mu + \boldsymbol{A}\xi \rangle \sim N(x^\top \mu, x^\top \boldsymbol{A}\boldsymbol{A}^\top x)$$

- **Uncertain Estimation**: **Find** A s.t. $AA^\top = \boldsymbol{\Sigma}$. $\Rightarrow f_\theta(x, \xi) \sim \langle \theta^*, x \rangle$.

- Similar concept in **[LLZ+22] (ICLR)** & [DLI+20, OWA+23a]

Ensemble Sampling as Special Case of Hypermodel Index Sampling

Sampling from Linear-Gaussian model $N(x^\top \mu, x^\top \boldsymbol{\Sigma} x)$, we could perform

Ensemble Sampling (# of models M)

$P_\xi = \mathcal{U}\{e_1, \ldots, e_M\}$ and $\theta = \boldsymbol{A} = [\tilde{\theta}_1, \ldots, \tilde{\theta}_M] \in \mathbb{R}^{d \times M}$, s.t. $\tilde{\theta}_m \sim N(\mu, \boldsymbol{\Sigma})$.

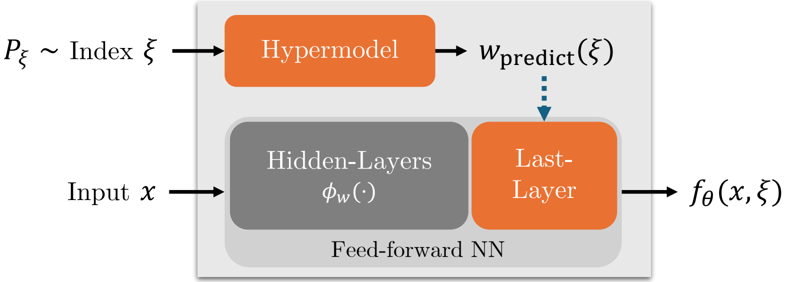
$$\xi \sim P_\xi \Rightarrow f_\theta(x, \xi) := \langle x, \boldsymbol{A}\xi \rangle \quad \text{where} \quad \boldsymbol{A}\xi \rangle \sim \mathcal{U}\{\tilde{\theta}_1, \ldots, \tilde{\theta}_M\}$$

Histogram approximation:

- $\tilde{\mu} = \mathbb{E}[\boldsymbol{A}\xi \mid \boldsymbol{A}] = \frac{1}{M} \sum_{i=1}^M \tilde{\theta}_i \to \mu$ as $M \uparrow$.
- $\mathrm{Cov}[\boldsymbol{A}\xi \mid \boldsymbol{A}] = \frac{1}{M} \sum_{i=1}^M (\tilde{\theta}_i - \tilde{\mu})(\tilde{\theta}_i - \tilde{\mu})^\top \to \boldsymbol{\Sigma}$ as $M \uparrow$.
- **Problem**: $M \uparrow$ leads to **unbounded computation**.

HyperAgent: Hypermodel for Feedforward Deep Networks

- **Base model**: DNN $\langle \phi_w(\cdot), w_{\text{predict}} \rangle$



- **Hypermodel**: [LXHL24] chooses $f_\theta(x, \xi) = \langle \phi_w(x), w_{\text{predict}}(\xi) \rangle$ with $w_{\text{predict}}(\xi) = A\xi + b$

$$f_\theta(x, \xi) = \underbrace{\langle \phi_w(x), b \rangle}_{\textbf{`mean'}\ \mu_\theta(x)} + \underbrace{\langle \phi_w(x), A\xi \rangle}_{\textbf{`variance'}\ \sigma_\theta(x, \xi)}$$

The degree of uncertainty

HyperAgent: Hypermodel for Deep RL

- **Base model** for **DQN-type value function**

$$f_\theta(s,a) = \langle \phi_w(s), \theta^{(a)} \rangle$$

with parameters $\theta = \{w, (\theta^{(a)} \in \mathbb{R}^d) : a \in \mathcal{A}\}$

Action-specific parameters for discrete action set $\mathcal{A}$

- **Hypermodel** for **randomized value function** depends on (s,a) and a **random** index $\xi \sim P_\xi$:

$$f_\theta(s,a,\xi) = \langle \phi_w(s), \underbrace{A^{(a)}\xi + b^{(a)}}_{\theta^{(a)}(\xi)} \rangle$$

Random index $\xi \sim P_\xi$

with parameters $\theta = \{w, (A^{(a)} \in \mathbb{R}^{d\times M}, b^{(a)}) : a \in \mathcal{A}\}$.

Action-specific parameters

- **Tabular** representation: $\phi_w(s)$ is fixed one-hot vector in $\mathbb{R}^{|\mathcal{S}|}$ where $d = |S|$. **(Unification!)**

HyperAgent: Seemless Integration to DDQN

Algorithm `HyperAgent` Framework

Input: Initial parameter θ_{init}, hypermodel f_θ with reference dist. P_ξ and perturbation dist. $P_{\mathbf{z}}$.

Init. $\theta = \theta^- = \theta_{\text{init}}$, train step $j = 0$ and buffer D

for each episode $k = 1, 2, \ldots$ **do**

  Sample index mapping $\xi_k \sim P_\xi$

  Set $t = 0$ and **Observe** $S_{k,0} \sim \rho$

  repeat

    Select $A_{k,t} = \arg\max_{a \in \mathcal{A}} f_\theta(S_{k,t}, a, \xi_k(S_{k,t}))$

    Observe $S_{k,t+1}$ **from environment** and $R_{k,t+1} = r(S_{k,t}, A_{k,t}, S_{k,t+1})$.

    Sample perturbation random vector $\mathbf{z}_{k,t+1} \sim P_{\mathbf{z}}$

    D.**add**$((S_{k,t}, A_{k,t}, R_{k,t+1}, S_{k,t+1}, \mathbf{z}_{k,t+1}))$

    Increment step counter $t \leftarrow t + 1$

    $\theta, \theta^-, j \leftarrow \texttt{update}(D, \theta, \theta^-, \xi^- = \xi_k, t, j)$

  until $S_{k,t} = s_{\text{terminal}}$

end for

HyperAgent: Objective for Generic Hypermodel (f_θ, P_ξ)

perturbation random vector

- For a transition tuple $d = (s, a, r, s', \boxed{\mathbf{z}}) \in D$ and given index ξ, the temporal difference (TD) error:

discounted factor

$$\ell^{\gamma,\sigma}(\boxed{\theta}; \boxed{\theta^-}, \boxed{\boldsymbol{\xi}^-}, \boxed{\xi}, d) = \left(f_\theta(s, a, \boxed{\xi}) - (r + \boxed{\sigma}\,\boxed{\xi}^\top \boxed{\mathbf{z}} + \boxed{\gamma} \max_{a' \in \mathcal{A}} f_{\theta^-}(s', a', \boxed{\boldsymbol{\xi}^-(s')})) \right)^2 \quad (2)$$

control the std of injected noise

target parameters, fixed here and updated in an outer loop

main parameters, optimization variables

- $\boxed{\boldsymbol{\xi}^-}$: the target index mapping s.t. $\boldsymbol{\xi}^-(s)$ one-to-one maps each state $s \in \mathcal{S}$ to a random vector from P_ξ, all of which are **independent** with $\boxed{\xi}$.

HyperAgent: Objective and Training

- Integrate ξ over Equation (2) yields objective $L^{\gamma,\sigma,\beta}$ where $\beta \geqslant 0$ is for the prior regularization

$$L^{\gamma,\sigma,\beta}(\theta;\theta^-,\boldsymbol{\xi}^-,D) = \mathbb{E}_{\xi\sim P_\xi}[\sum_{d\in D} \frac{1}{|D|}\ell^{\gamma,\sigma}(\theta;\theta^-,\boldsymbol{\xi}^-,\xi,d)] + \frac{\beta}{|D|}\|\theta\|^2 \tag{3}$$

- Optimize **main objective Equation (3)** using **mini-batch SGD (default Adam)**, i.e., sampled loss

$$\tilde{L}(\theta;\theta^-,\boldsymbol{\xi}^-,\underbrace{\tilde{D}}_{\text{a batch of data } \tilde{D} \text{ sampled from } D}) = \frac{1}{|\tilde{\Xi}|}\sum_{\xi\in \underbrace{\tilde{\Xi}}_{\text{a batch of indices } \tilde{\Xi} \text{ sampled from } P_\xi}} \left(\sum_{d\in\tilde{D}} \frac{1}{|\tilde{D}|}\ell^{\gamma,\sigma}(\theta;\theta^-,\boldsymbol{\xi}^-,\xi,d)\right) + \frac{\beta}{|D|}\|\theta\|^2 \tag{4}$$

- **Update** the main parameters θ in each step according to Equation (4), and updates the target parameters θ^- periodically with less frequency. $\Rightarrow$ **Bounded per-step computation**.

Outline

Motivation: RL under Resource Constraints

Existing solution and their limitations

- Our Contributions

HyperAgent for Efficient RL [LXHL24] **(ICML)**

- Theoretical insights with tabular representation
- Empirical evidence on efficient deep exploration with deep RL
- Reduce **sequential posterior approx.** to sequential random projection

GPT-HyperAgent for Continual Content Moderation

Theoretical Understanding via Unified Representation and Algorithm

- **Tabular** representation: $\phi_w(s)$ is fixed one-hot vector in $\mathbb{R}^{|\mathcal{S}|}$ where $d = |S|$. **(Unification!)**
- **Tabular** `HyperAgent`: short notations

$$
\begin{aligned}
f_\theta(s, a, \xi) &= \langle \phi_w(s), A^{(a)}\xi + b^{(a)} \rangle \\
&= \langle \underbrace{(A^{(a)})^\top \phi_w(s)}_{\tilde{m}_{sa}} + \underbrace{(b^{(a)})^\top \phi_w(s)}_{\mu_{sa}}, \xi \rangle
\end{aligned}
$$

- Parameters in k-th episode $\theta_k = (\mu_{k,sa}, \tilde{m}_{k,sa} \in \mathbb{R}^M, \forall (s,a) \in \mathcal{S} \times \mathcal{A})$.
- $\phi_w(s)$ **fixed mapping**, e.g. tabular and linear FA.
- $\Rightarrow$ Equation (3) of **HyperAgent permits closed-form solution.**
  - HyperDQN [LLZ+22] & ENN-DQN[OWA+23b] can **not** derive closed-form solution.

Insights from closed-form solution

- **Incremental update** with computation complexity $O(M)$:

Perturbation random vector

$$\tilde{m}_{k,sa} = \frac{(N_{k-1,sa}+\beta)\,\tilde{m}_{k-1,sa} + \sum_{t\in E_{k-1,sa}} \sigma\,\mathbf{z}_{\ell,t+1}}{(N_{k,sa}+\beta)} \in \mathbb{R}^M \tag{5}$$

Set of timesteps encoutering (s,a) in episode $k-1$

Visitation counts of (s,a) up to episode k

Lemma 1 (Sequential posterior approximation via incremental update).

For $\tilde{m}_k$ recursively defined in Equation (5) with $\mathbf{z} \sim \mathcal{U}(\mathbb{S}^{M-1})$. For any $k \geqslant 1$, define the good event of ε-approximation

$$\mathcal{G}_{k,sa}(\varepsilon) := \left\{ \|\tilde{m}_{k,sa}\|^2 \in \left((1-\varepsilon)\frac{\sigma^2}{N_{k,sa}+\beta}, (1+\varepsilon)\frac{\sigma^2}{N_{k,sa}+\beta} \right) \right\}.$$

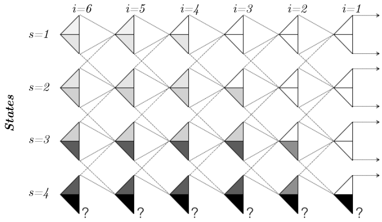
*The **joint event** $\cap_{(s,a)\in\mathcal{S}\times\mathcal{A}} \cap_{k=1}^{K} \mathcal{G}_{k,sa}(\varepsilon)$ holds w.p. at least $1-\delta$ if $M \simeq \varepsilon^{-2}\log(SAHK/\delta)$.*

Insights from closed-form solution

Stochastic Bellman Operator F_k^γ induced by Equation (3) w. $\theta = \theta_k^{(i+1)}, \theta^- = \theta_k^{(i)}$ **iteratively**:

$$f_{\theta_k^{(i+1)},\xi_k} = F_k^\gamma f_{\theta_k^{(i)},\xi_k} \approx (r_{sa} + \gamma \langle V_{f_{\theta_k^{(i)},\xi_k}}, \underbrace{\hat{P}_{k,sa}}_{\text{Empirical transition}} \rangle) + \underbrace{\tilde{m}_{k,sa}^\top \xi_k(s)}_{\text{"Randomized bonus"} \propto \sqrt{\frac{1}{N_{k,sa}}}}, \tag{6}$$

where $f_{\theta,\xi^-}(s,a) = f_\theta(s,a,\xi^-(s))$ and $V_Q(s) := \max_a Q(s,a), \forall s$ is the greedy value w.r.t. Q.



Setup: $N_{k,(4,\searrow)} = 1$. Other (s,a) almost **infinite data**.

(1) Propagation of uncertainty from later time period to earlier time period due to **iterative applying** F_k^γ.

(2) Darker shade indicates **higher** degree of uncertainty.

(3) Incentivize deep exploration.

Outline

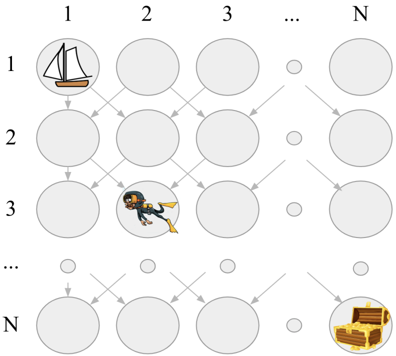
Motivation: RL under Resource Constraints

Existing solution and their limitations
- Our Contributions

HyperAgent for Efficient RL [LXHL24] **(ICML)**
- Theoretical insights with tabular representation
- Empirical evidence on efficient deep exploration with deep RL
- Reduce **sequential posterior approx.** to sequential random projection

GPT-HyperAgent for Continual Content Moderation

Simple Illustration for Deep Exploration: DeepSea Environment

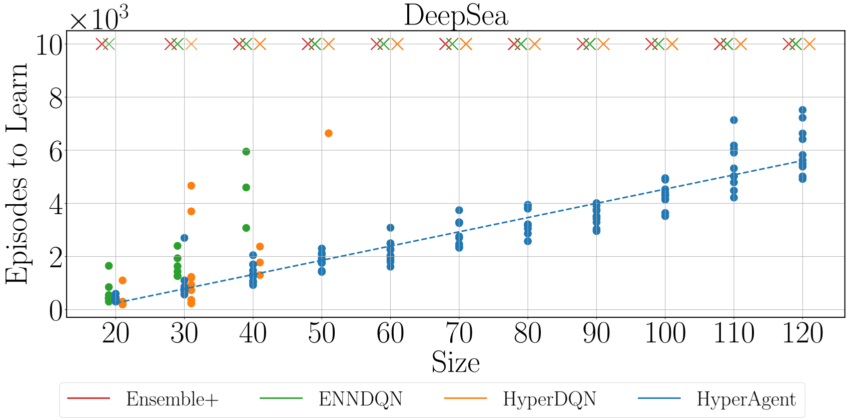


DeepSea: The agent receives a reward of 0 for ↙, and a penalty of $-(0.01/N)$ for ↘, where N denotes the size of DeepSea. The agent will earn a reward of 1 upon reaching the lower-right corner but she do NOT know in advances whether there is a reward until reaching the goal.

exploration method	expected episodes to learn
optimal	$\Theta(N)$
pure exploitation	∞
dithering (ϵ-greedy)	$\Theta(2^N)$
optimistic	$\Theta(N)$
randomized	$\Theta(N)$

Expected number of episodes required to learn an optimal policy for DeepSea with size N.
Optimistic: optimism in the face of uncertainty (**OFU**);
Randomized: randomizing the belief of the environment, e.g. **Posterior sampling**

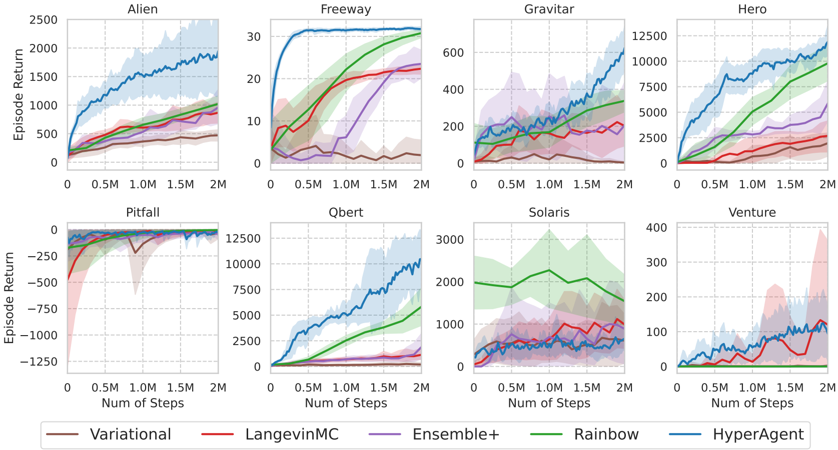
HyperAgent: Efficiency in benchmarks (DeepSea)



Comparison with **Ensemble+** [OAC18, OVRRW19], **HyperDQN** [LLZ+22], **ENN-DQN**[OWA+23b].

- ✓ **Scalable** as size $N \uparrow$. State representation: one-hot vector in high-dimension $\mathbb{R}^N$.
- ✓ **Data efficient**: `HyperAgent` the only and first achieving **optimal episode complexity** $\Theta(N)$.

HyperAgent: comparison with other posterior approximation methods



Comparison on approximate posterior sampling methods: **Variational approximation** (SANE [AL21]), **Langevin Monte-Carlo** (AdamLMCDQN [ILX+24]) and **Ensemble+** [OAC18, OVRRW19]

Outline

Motivation: RL under Resource Constraints

Existing solution and their limitations

Our Contributions

HyperAgent for Efficient RL [LXHL24] **(ICML)**

Theoretical insights with tabular representation

Empirical evidence on efficient deep exploration with deep RL

Reduce **sequential posterior approx.** to sequential random projection

GPT-HyperAgent for Continual Content Moderation

Step 1: Rewrite incremental update on $\tilde{m}_{k,sa}$

- E_ℓ: the collection of time steps in episode ℓ.
- $E_{\ell,sa}$: the collection of time steps in episode ℓ encountering state-action pair (s,a).
- Define a sequence of indicator variables $x_{\ell,t} = \mathbb{1}_{t \in E_{\ell,sa}}$. Note

$$\sum_{\ell=1}^{k-1} \sum_{t \in E_\ell} x_{\ell,t}^2 = N_{k,sa}$$

- Define short notations $\mathbf{z}_0 = \mathbf{z}_{0,sa}$ and $x_0 = \sqrt{\beta}$. Let $\beta = \sigma^2 / \sigma_0^2$. Equation (5) now becomes

$$\frac{(N_{k,sa} + \beta)}{\sigma} \tilde{m}_{k,sa} = x_0 \mathbf{z}_0 + \sum_{\ell=1}^{k-1} \sum_{t \in E_\ell} x_{\ell,t} \mathbf{z}_{\ell,t+1} \tag{7}$$

- **Lemma 1 $\Rightarrow$ w.h.p. Equation (8) holds for all $(s,a) \in \mathcal{S} \times \mathcal{A}$ and $k \in [K]$ simultaneously**:

$$(1-\varepsilon)(x_0^2 + \sum_{\ell=1}^{k-1} \sum_{t \in E_\ell} x_{t,\ell}^2) \leqslant \|x_0 \mathbf{z}_0 + \sum_{\ell=1}^{k-1} \sum_{t \in E_\ell} x_{\ell,t} \mathbf{z}_{\ell,t+1}\|^2 \leqslant (1+\varepsilon)(x_0^2 + \sum_{\ell=1}^{k-1} \sum_{t \in E_\ell} x_{\ell,t}^2) \tag{8}$$

Classical JL for random projection

- Try to relate Equation (8) to the classical **Johnson–Lindenstrauss (JL)** lemma:

$$\text{Consider} \quad \Pi = (\mathbf{z}_1, \ldots, \mathbf{z}_d) \in \mathbb{R}^{M \times d}, \quad x = (x_1, \ldots, x_d)^\top \in \mathbb{R}^d, \quad \text{then} \quad \Pi x = \sum_{i=1}^{d} x_i \mathbf{z}_i$$

Lemma 2 (Distributional JL lemma [JL84]).

For any $0 < \varepsilon, \delta \leqslant 1/2$ and $d \geqslant 1$ there exists a distribution $\mathcal{D}_{\varepsilon,\delta}$ on $\mathbb{R}^{M \times d}$ for $M = O\left(\varepsilon^{-2} \log(1/\delta)\right)$ such that for any $x \in \mathbb{R}^d$

$$\mathbb{P}_{\Pi \sim \mathcal{D}_{\varepsilon,\delta}} \left(\|\Pi x\|_2^2 \notin \left[(1-\varepsilon)\|x\|_2^2, (1+\varepsilon)\|x\|_2^2 \right] \right) < \delta$$

- **Existing JL analysis** based on the **assumption**: x fixed non-random or the projection matrix Π is generated **independently** with the data x, i.e.

$$\Pi := (\mathbf{z}_1, \ldots, \mathbf{z}_d) \perp\!\!\!\perp x := (x_1, \ldots, x_d).$$

Step 2: Indentify dependence structure

Sequential dependence structure in `HyperAgent` when interacting with environment is that

- E_ℓ: the collection of time steps in episode ℓ.
- $E_{\ell,sa}$: the collection of time steps in episode ℓ
- $x_{\ell,t} = \mathbb{1}_{t \in E_{\ell,sa}}$ is **dependent** on the environmental and algorithmic randomness in all previous time steps:

$$\mathbf{z}_0, (x_{1,t'}, \mathbf{z}_{1,t'+1})_{t' \in E_1}, (x_{2,t'}, \mathbf{z}_{2,t'+1})_{t' \in E_2}, \ldots, (x_{\ell,t'}, \mathbf{z}_{\ell,t'+1})_{t' < t};$$

- $\mathbf{z}_{\ell,t+1}$ is **independent** of the environmental and algorithmic randomness in all previous time steps:

$$\mathbf{z}_0, (x_{1,t'}, \mathbf{z}_{1,t'+1})_{t' \in E_1}, (x_{2,t'}, \mathbf{z}_{2,t'+1})_{t' \in E_2} \ldots, (x_{\ell,t'+1}, \mathbf{z}_{\ell,t'+1})_{t' < t}, x_{\ell,t},$$

Difficulty and Novelty in the Mathemtical Analysis: No Prior Art

$$
\begin{tikzcd}
& \text{Time } t=1 & \text{Time } t=2 & \text{Time } t=3 & & \text{Time } t & \text{Time } t+1 \\
& x_1 \arrow[r] & x_2 \arrow[r] & x_3 \arrow[r] & \cdots \arrow[r] & x_t \arrow[r] & x_{t+1} \\
\mathbf{z}_0 \arrow[ur] & \mathbf{z}_1 \arrow[ur] & \mathbf{z}_2 \arrow[ur] & \mathbf{z}_3 \arrow[ur] & \cdots \arrow[ur] & \mathbf{z}_t \arrow[ur] &
\end{tikzcd}
$$

Sequential dependence of **high-dimensional** R.V. due to the adaptive nature of sequential decision-making.

Difficulty: **(1)** Conditioned on x_t, $(\mathbf{z}_s)_{s<t}$ **loss** their independence; **(2) No** characterization on $P_{(\mathbf{z}_s)_{s<t}|x_t}$.
$\Rightarrow$ **Traditional analysis of random projection cannot handle sequential dependence** [Li24a].

First probability tool for **sequential random projection**. [Li24a]

- A **non-trivial martingale extension** of the **Johnson–Lindenstrauss (JL)**.
- **Technical novelty**: a careful construction of **stopped process** with non-trivial application of **'method of mixtures'** in self-normalized martingale.

Sequential Random Projection

Theorem 1 (Sequential random projection in adaptive processes [Li24a]).

– *Let* $\varepsilon \in (0,1)$ *be fixed and* $(\mathcal{F}_t)_{t \geqslant 0}$ *be a filtration. Let* $\mathbf{z}_0 \in \mathbb{R}^M$ *be an* $\mathcal{F}_0$*-measurable random vector satisfies* $\mathbb{E}[\|\mathbf{z}_0\|^2] = 1$ *and* $|\|\mathbf{z}_0\|^2 - 1| \leqslant (\varepsilon/2)$.
– *Let* $(\mathbf{z}_t)_{t \geqslant 1} \subset \mathbb{R}^M$ *be a stochastic process adapted to filtration* $(\mathcal{F}_t)_{t \geqslant 1}$ *such that it is* $\sqrt{c_0/M}$*-sub-Gaussian and each* $\mathbf{z}_t$ *is unit-norm.* – *Let* $(x_t)_{t \geqslant 1} \subset \mathbb{R}$ *be a stochastic process adapted to filtration* $(\mathcal{F}_{t-1})_{t \geqslant 1}$ *such that it is* c_x*-bounded. Here,* c_0 *and* c_x *are absolute constants.*
– *For any fixed* $x_0 \in \mathbb{R}$, *if the following condition is satisfied*

$$M \geqslant \frac{16 c_0 (1+\varepsilon)}{\varepsilon^2} \left(\log\left(\frac{1}{\delta}\right) + \log\left(1 + \frac{c_x T}{x_0^2}\right)\right),$$

we have, with probability at least $1-\delta$

$$\forall t \in \{0, 1, \ldots, T\}, \quad (1-\varepsilon)\left(\sum_{i=0}^{t} x_i^2\right) \leqslant \|\sum_{i=0}^{t} x_i \mathbf{z}_i\|^2 \leqslant (1+\varepsilon)\left(\sum_{i=0}^{t} x_i^2\right).$$

Outline

Motivation: RL under Resource Constraints

Existing solution and their limitations

- Our Contributions

HyperAgent for Efficient RL [LXHL24] **(ICML)**

- Theoretical insights with tabular representation
- Empirical evidence on efficient deep exploration with deep RL
- Reduce **sequential posterior approx.** to sequential random projection

GPT-HyperAgent for Continual Content Moderation

Distributed Continual Content Moderation

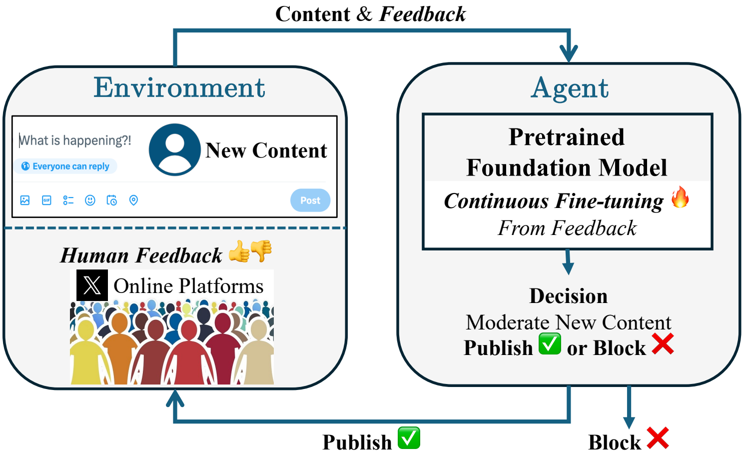
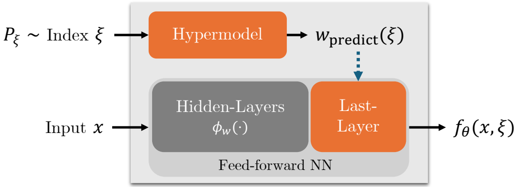


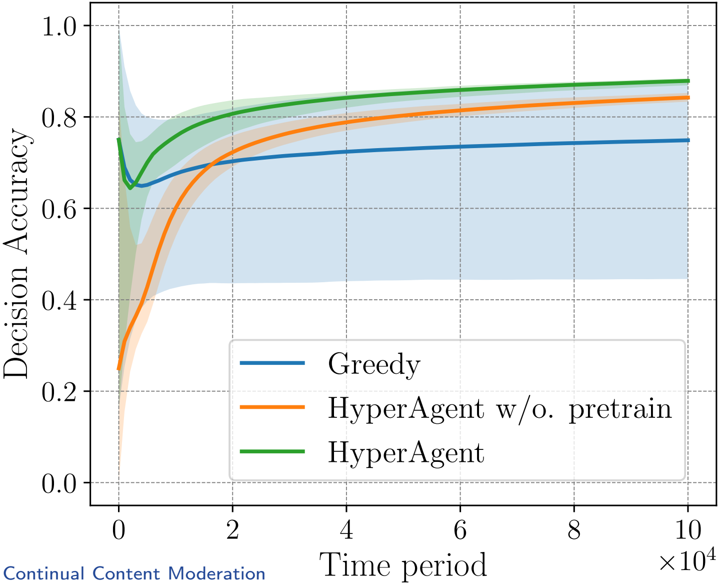
Figure: Only-published content is sent to the crowd-sourced feedback system.

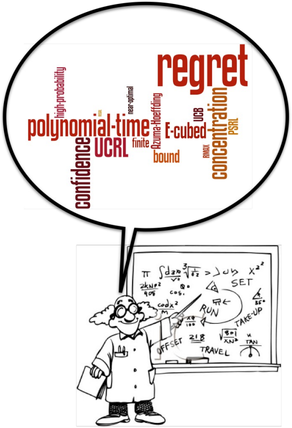
GPT-HyperAgent



- **Context**: text **content** from Hugging Face (HF): '`ucberkeley-dlab/measuring-hate-speech`' coming sequentially.
- AI Moderator: decide to publish or not according to regression output and a threshold.
- **Human Feedback**: In our experiments, this is simulated by a harmfulness attribute in the dataset, translating to binary feedback.
- Training: **GPT-2** with **Hypermodel** for the last layer. Regression on the feedback.

Distributed Continual Content Moderation





Fulfilling
the Promise of RL

References I

[AJZ23] Alekh Agarwal, Yujia Jin, and Tong Zhang. Vo q l: Towards optimal regret in model-free rl with nonlinear function approximation. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 987–1063. PMLR, 2023.

[AL21] Siddharth Aravindan and Wee Sun Lee. State-aware variational thompson sampling for deep q-networks. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, pages 124–132, 2021.

[DAHVR24] Vikranth Dwaracherla, Seyed Mohammad Asghari, Botao Hao, and Benjamin Van Roy. Efficient exploration for llms. *arXiv preprint arXiv:2402.00396*, 2024.

[DKL+21] Simon Du, Sham Kakade, Jason Lee, Shachar Lovett, Gaurav Mahajan, Wen Sun, and Ruosong Wang. Bilinear classes: A structural framework for provable generalization in rl. In *International Conference on Machine Learning*, pages 2826–2836. PMLR, 2021.

[DLI+20] Vikranth Dwaracherla, Xiuyuan Lu, Morteza Ibrahimi, Ian Osband, Zheng Wen, and Benjamin Van Roy. Hypermodels for exploration. In *International Conference on Learning Representations*, 2020.

References II

[DMM+22] Chris Dann, Yishay Mansour, Mehryar Mohri, Ayush Sekhari, and Karthik Sridharan. Guarantees for epsilon-greedy reinforcement learning with function approximation. In *International conference on machine learning*, pages 4666–4689. PMLR, 2022.

[DMZZ21] Christoph Dann, Mehryar Mohri, Tong Zhang, and Julian Zimmert. A provably efficient model-free posterior sampling method for episodic reinforcement learning. *Advances in Neural Information Processing Systems*, 34:12040–12051, 2021.

[FKQR21] Dylan J Foster, Sham M Kakade, Jian Qian, and Alexander Rakhlin. The statistical complexity of interactive decision making. *arXiv preprint arXiv:2112.13487*, 2021.

[HMVH+18] Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

References III

[ICN+21] Haque Ishfaq, Qiwen Cui, Viet Nguyen, Alex Ayoub, Zhuoran Yang, Zhaoran Wang, Doina Precup, and Lin Yang. Randomized exploration in reinforcement learning with general value function approximation. In *International Conference on Machine Learning*, pages 4607–4616. PMLR, 2021.

[ILX+24] Haque Ishfaq, Qingfeng Lan, Pan Xu, A. Rupam Mahmood, Doina Precup, Anima Anandkumar, and Kamyar Azizzadenesheli. Provable and practical: Efficient exploration in reinforcement learning via langevin monte carlo. In *The Twelfth International Conference on Learning Representations*, 2024.

[JKA+17] Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. Contextual decision processes with low bellman rank are pac-learnable. In *International Conference on Machine Learning*, pages 1704–1713. PMLR, 2017.

[JL84] William B Johnson and Joram Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. In *Conference on Modern Analysis and Probability*, volume 26, pages 189–206. American Mathematical Society, 1984.

[JLM21] Chi Jin, Qinghua Liu, and Sobhan Miryoosefi. Bellman eluder dimension: New rich classes of rl problems, and sample-efficient algorithms. *Advances in neural information processing systems*, 34:13406–13418, 2021.

[Kak03] Sham Machandranath Kakade. *On the sample complexity of reinforcement learning*. University of London, University College London (United Kingdom), 2003.

[Li24a] Yingru Li. Probability Tools for Sequential Random Projection, 2024. arXiv: 2402.14026.

[Li24b] Yingru Li. Simple, unified analysis of Johnson-Lindenstrauss with applications, 2024. arXiv: 2402.10232.

[LL24] Yingru Li and Zhiquan Luo. Prior-dependent analysis of posterior sampling reinforcement learning with function approximation. In *International Conference on Artificial Intelligence and Statistics*, pages 559–567. PMLR, 2024.

References V

[LLX+23] Zhihan Liu, Miao Lu, Wei Xiong, Han Zhong, Hao Hu, Shenao Zhang, Sirui Zheng, Zhuoran Yang, and Zhaoran Wang. Maximize to explore: One objective function fusing estimation, planning, and exploration. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[LLZ+22] Ziniu Li, Yingru Li, Yushun Zhang, Tong Zhang, and Zhi-Quan Luo. HyperDQN: A randomized exploration method for deep reinforcement learning. In *International Conference on Learning Representations*, 2022.

[LXHL24] Yingru Li, Jiawei Xu, Lei Han, and Zhi-Quan Luo. Q-Star Meets Scalable Posterior Sampling: Bridging Theory and Practice via HyperAgent. In *Forty-first International Conference on Machine Learning*, Proceedings of Machine Learning Research, 2024.

[OAC18] Ian Osband, John Aslanides, and Albin Cassirer. Randomized prior functions for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 31, 2018.

[OBPVR16] Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. *Advances in neural information processing systems*, 29, 2016.

References VI

[OVR17] Ian Osband and Benjamin Van Roy. Why is posterior sampling better than optimism for reinforcement learning? In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 2701–2710. JMLR.org, 2017.

[OVRRW19] Ian Osband, Benjamin Van Roy, Daniel J. Russo, and Zheng Wen. Deep exploration via randomized value functions. *Journal of Machine Learning Research*, 20(124):1–62, 2019.

[OWA+23a] Ian Osband, Zheng Wen, Seyed Mohammad Asghari, Vikranth Dwaracherla, Morteza Ibrahimi, Xiuyuan Lu, and Benjamin Van Roy. Epistemic neural networks. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[OWA+23b] Ian Osband, Zheng Wen, Seyed Mohammad Asghari, Vikranth Dwaracherla, Morteza Ibrahimi, Xiuyuan Lu, and Benjamin Van Roy. Approximate thompson sampling via epistemic neural networks. *arXiv preprint arXiv:2302.09205*, 2023.

[RVR16] Daniel Russo and Benjamin Van Roy. An information-theoretic analysis of thompson sampling. *The Journal of Machine Learning Research*, 17(1):2442–2471, 2016.

[RVRK+18] Daniel J Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, Zheng Wen, et al. A tutorial on thompson sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96, 2018.

[SCC+23] Max Schwarzer, Johan Samir Obando Ceron, Aaron Courville, Marc G Bellemare, Rishabh Agarwal, and Pablo Samuel Castro. Bigger, better, faster: Human-level atari with human-level efficiency. In *International Conference on Machine Learning*, pages 30365–30380. PMLR, 2023.

[Str00] Malcolm Strens. A bayesian framework for reinforcement learning. In *ICML*, volume 2000, pages 943–950, 2000.

[Str07] Alexander L Strehl. *Probably approximately correct (PAC) exploration in reinforcement learning*. PhD thesis, Rutgers University-Graduate School-New Brunswick, 2007.

[VHGS16] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

References VIII

[WSY20] Ruosong Wang, Russ R Salakhutdinov, and Lin Yang. Reinforcement learning with general value function approximation: Provably efficient approach via bounded eluder dimension. *Advances in Neural Information Processing Systems*, 33:6123–6135, 2020.

[XZM+22] Pan Xu, Hongkai Zheng, Eric V Mazumdar, Kamyar Azizzadenesheli, and Animashree Anandkumar. Langevin monte carlo for contextual bandits. In *International Conference on Machine Learning*, pages 24830–24850. PMLR, 2022.

[Zha22] Tong Zhang. Feel-good thompson sampling for contextual bandits and reinforcement learning. *SIAM Journal on Mathematics of Data Science*, 4(2):834–857, 2022.

[ZVR23] Zheqing Zhu and Benjamin Van Roy. Deep exploration for recommendation systems. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 963–970, 2023.

[ZXZ+22] Han Zhong, Wei Xiong, Sirui Zheng, Liwei Wang, Zhaoran Wang, Zhuoran Yang, and Tong Zhang. Gec: A unified framework for interactive decision making in mdp, pomdp, and beyond. *arXiv preprint arXiv:2211.01962*, 2022.