南京航空航天大學
Nanjing University of Aeronautics and Astronautics

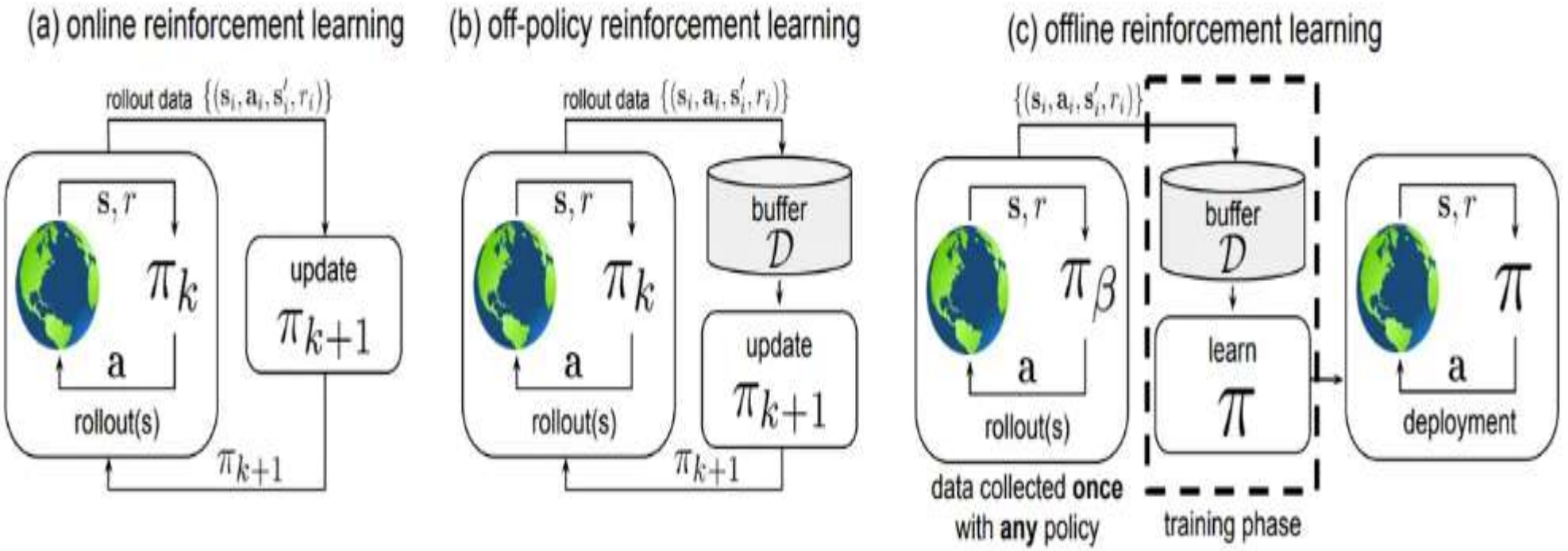
模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

Deep Reinforcement Learning from Human Preferences

NIPS | 2017

Background

- **Reinfocement learning**



Background

- **Preference-based reinforcement learning (PbRL)**

does not rely directly on numerical rewards provided by the environment. Instead, it learns a policy through human or external feedback in the form of preference information over behaviors. This approach alleviates the need for precisely designed reward functions.

- **pipelines:**
  - •Collecting human preference feedback by comparing pairs of trajectory segments,
  - •Training a reward model, usually using a pairwise loss,
  - •Optimizing the policy using the trained reward model, which can be done with either on-policy or offline methods.

The pairwise loss is typically formulated as follows:

$$\mathcal{L}_{\text{pref}} = -\log\left(\frac{e^{R(\tau_1)}}{e^{R(\tau_1)} + e^{R(\tau_2)}}\right) \tag{5}$$

- **Motivations**
  - 将该方法适配到深度强化学习并适用于高难度任务

Preliminary

Trajectory Segment:

- A short sequence:

$$\tau = \left((o_0, a_0), (o_1, a_1), \ldots, (o_{k-1}, a_{k-1})\right)$$

- Human compares segments:

$$\tau_1 \succ \tau_2 \quad \text{means human prefers } \tau_1.$$

- **评价指标（策略）**
  - 定量评价
    - 当环境有已知的价值函数，直接计算总奖励进行比较。
  - 定性评价
    - 人来看看学出来的策略表不表现得让人满意。

Methodology

- **Main Procedures**
  - get a set of trajectory $\{\tau^0, \tau^1, ..., \tau^i\}$ through training policy π
  - select pairs of segments $\{\sigma 1, \sigma 2\}$ from trajectory and send to human for comparison
  - train reward function
  - use reward function to update policy

[Policy π] --(generate)--> [Trajectories] --(select pairs)--> [Human Comparison]
↑ ↓
[Reward Model r̂] <---(fit)--- [Preference Data] <---

- **Policy update**

the authors use A2C to play Atari games, and TPRO to perform simulated robotics tasks.

Methodology

- **Preference Elicitation**

  The human overseer is given a visualization of two trajectory segments, in the form of short movie clips. In all of our experiments, these clips are between 1 and 2 seconds long.

  The human judgments are recorded in a database D of triples ($\sigma 1$, $\sigma 2$, μ)

- **Fitting the Reward Function**

  We can interpret a reward function estimate $\hat{r}$ as a preference-predictor. the procedure of reward model traning can be concluded as:

  - Use Bradley-Terry model to estimate reward based on observation and action
  - obtain optimization objective via B-T model
  - create deep neural network for reward model
  - training

Methodology

$$\hat{P}\left[\sigma^1 \succ \sigma^2\right] = \frac{\exp \sum \hat{r}\left(o_t^1, a_t^1\right)}{\exp \sum \hat{r}\left(o_t^1, a_t^1\right) + \exp \sum \hat{r}\left(o_t^2, a_t^2\right)}. \tag{1}$$

$$\operatorname{loss}(\hat{r}) = -\sum_{(\sigma^1, \sigma^2, \mu) \in \mathcal{D}} \mu(1) \log \hat{P}\left[\sigma^1 \succ \sigma^2\right] + \mu(2) \log \hat{P}\left[\sigma^2 \succ \sigma^1\right].$$

- **Selecting Queries**

The method selects trajectory pairs for human comparison by measuring uncertainty: it chooses pairs where an ensemble of reward models shows the highest disagreement.

Methodology

- **Special setting**
  - **Ensemble reward prediction(multi-model and independent sampling)**
  - **add L2 regularization to avoid overfit**
  - **add noise(10%) to reward model by supposing human can make mistake**

$$P_{\text{final}}(\text{segment 1 preferred}) = (1-p) \times \frac{e^{R_1}}{e^{R_1} + e^{R_2}} + p \times 0.5$$

Experiments

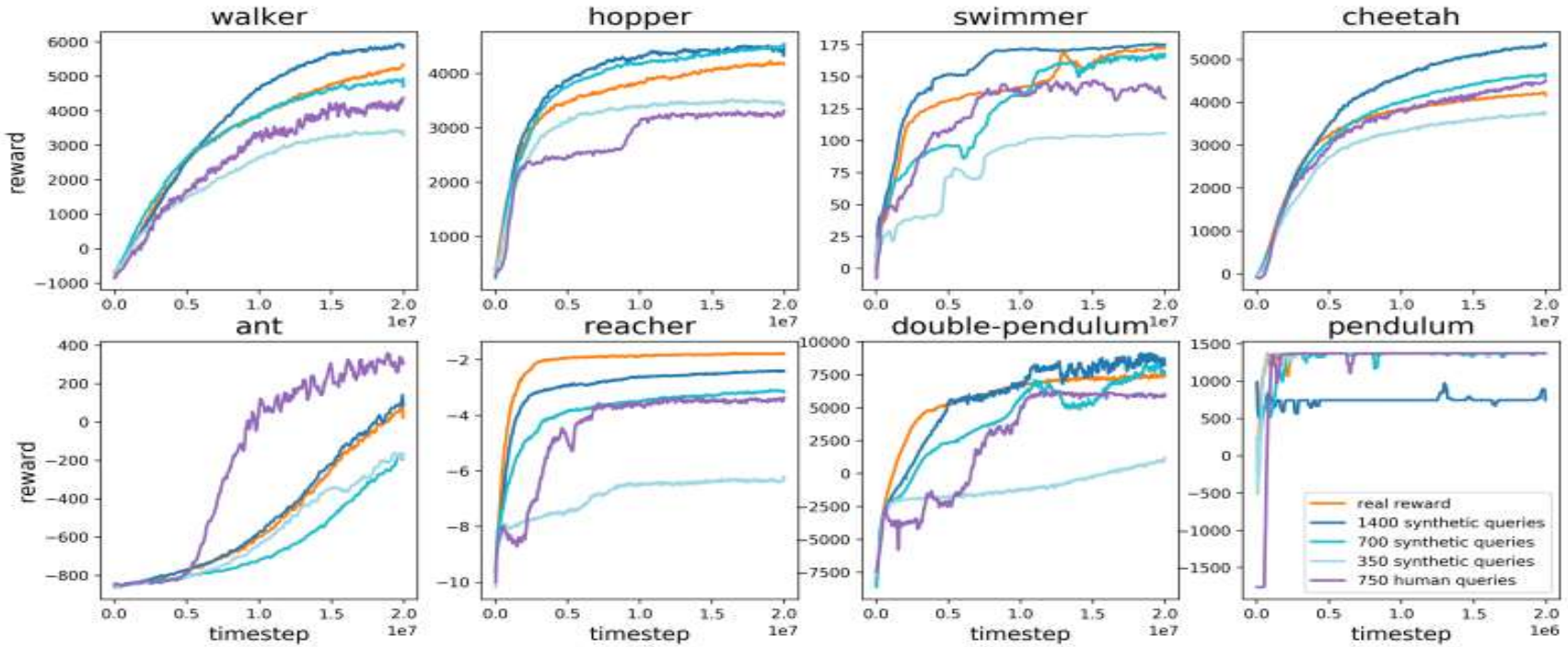


Figure 1: Results on MuJoCo simulated robotics as measured on the tasks' true reward. We compare our method using real human feedback (purple), our method using synthetic feedback provided by an oracle (shades of blue), and reinforcement learning using the true reward function (orange). All curves are the average of 5 runs, except for the real human feedback, which is a single run, and each point is the average reward over five consecutive batches. For Reacher and Cheetah feedback was provided by an author due to time constraints. For all other tasks, feedback was provided by contractors unfamiliar with the environments and with our algorithm. The irregular progress on Hopper is due to one contractor deviating from the typical labeling schedule.

Experiments

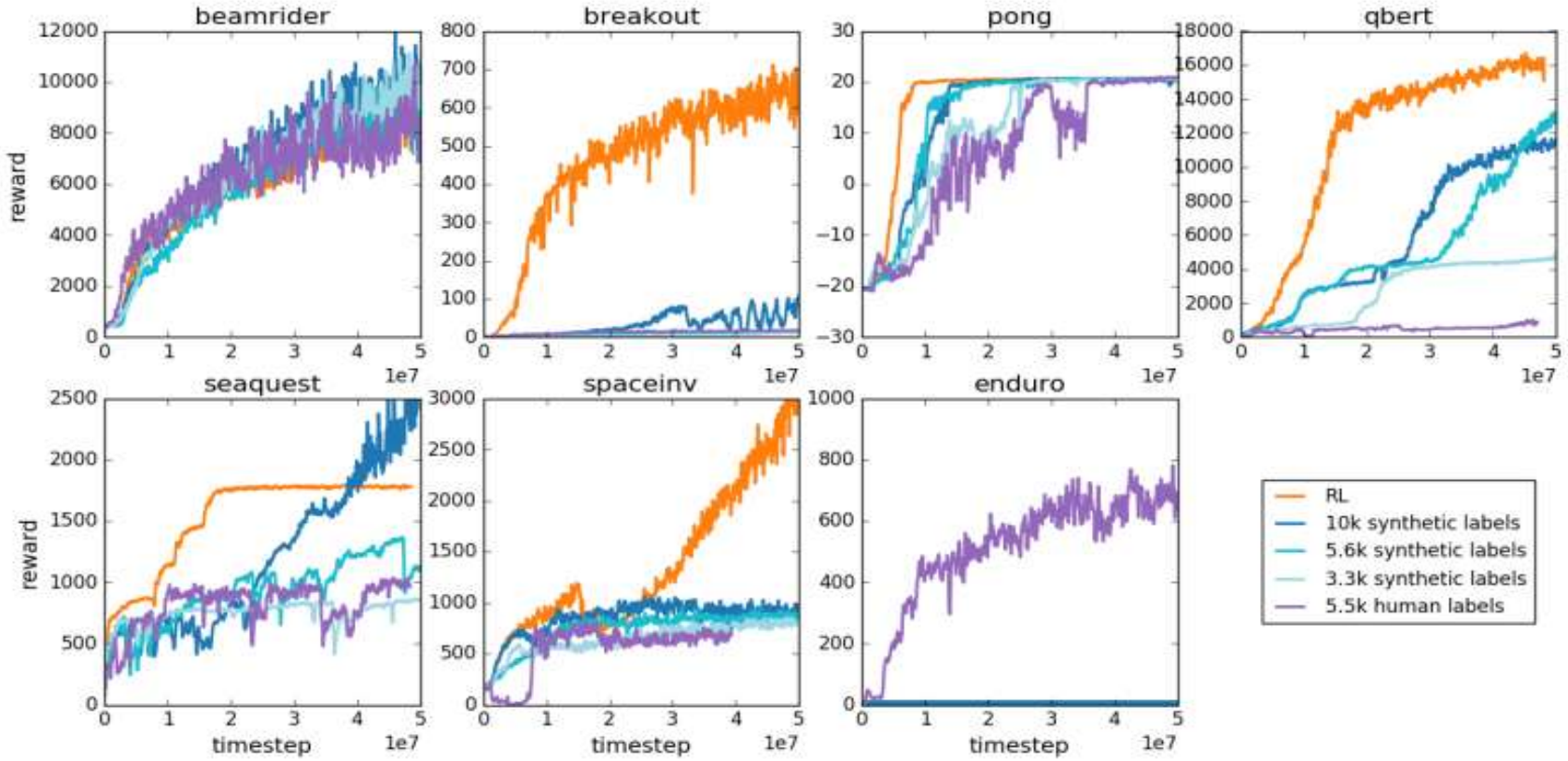


Figure 2: Results on Atari games as measured on the tasks' true reward. We compare our method using real human feedback (purple), our method using synthetic feedback provided by an oracle (shades of blue), and reinforcement learning using the true reward function (orange). All curves are the average of 3 runs, except for the real human feedback which is a single run, and each point is the average reward over about 150,000 consecutive frames.

Improvement

- 动机
  - 场景变化：若存在设计好的奖励模型，但奖励模型未知，并且采集的离线数据中有部分轨迹存在奖励。
    - 将B-T建模调整为基于已知奖励片段的B-T模型

  - query 方法提升：作者使用了奖励预测的方差来作为query的依据，能否优化query方法？

Improvement

Query流程简述（离线采样）

1. **轨迹片段对生成**
   从离线轨迹数据集中直接截取长度为 k 的轨迹片段，随机或基于规则组成轨迹片段对 (σ_i, σ_j)。
2. **奖励估计器预测**
   使用奖励模型集（ensemble）中的每个预测器，对轨迹片段计算奖励总和：

$$R(\sigma) = \sum_{t=0}^{k-1} \hat{r}(o_t, a_t)$$

3. **偏好概率计算**
   对每个预测器，计算轨迹片段偏好概率：

$$P(\sigma_i \succ \sigma_j) = \frac{\exp(R(\sigma_i))}{\exp(R(\sigma_i)) + \exp(R(\sigma_j))}$$

4. **方差估计**
   计算所有预测器对偏好概率的方差，作为轨迹对的不确定性指标：

$$\mathrm{Var}\big(P(\sigma_i \succ \sigma_j)\big)$$

5. **多样性约束**
   对方差最高的前N个轨迹片段对，利用轨迹距离（如欧氏距离）进行聚类或距离采样，确保选出的轨迹对覆盖不同状态空间区域。
6. **人工反馈查询**
   将筛选后的轨迹片段对以短视频形式呈现给人类，收集偏好标签用于后续训练。

Improvement

- 基于部分奖励轨迹的偏好强化
  - 有奖励轨迹微调
  - 有奖励轨迹数据增强（直接当成正确的样本加入训练）
  - 策略微调