Email Address:

zp2130@caa.columbia.edu (mailto:zp2130@caa.columbia.edu)

p@caa.columbia.edu (mailto:p@caa.columbia.edu)

Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation

Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation (https://blogs.cuit.columbia.edu/zp2130/files/2019/04/hierarchical-deep-reinforcement-learning-integrating-temporal-abstraction-and-intrinsic-motivation.pdf)

当环境给的奖励少而延迟时，论文给出了一个解决方案：agent至始至终只有一个，但分两个阶段：1总控器阶段，选goal，2控制器，根据当前state和goal，输出action，critic判断goal是否完成或达到终态。重复1,2。总控器选一个新的goal，控制器再输出action，依次类推。我理解它把环境“分”出N个时序上的小环境，与每个小环境对应1个goal。agent实体在这种环境下可以等效为一个点。

```
The key is that the policy over goals π_g which makes expected Q-value with discounting maximum is the policy which the agent chooses, i.e., if the goal sequence g1-g3-g2-... 's Q-value is the maximum value among that of all kinds of goal sequences, the agent should assign goal1 firstly, goal3 secondly, then goal2, ...
```

$$Q_2^*(s,g) = max_{\pi_g} E\left[\sum_{t'=t}^{t+N} f_{t'} + \gamma\ max_{g'} Q_2^*(s_{t+N}, g')\,|s_t = s, g_t = g, \pi_g\right]$$

以游戏为例：要走很久才能拿到钥匙，拿到钥匙后再走很久开门，所以说环境给agent的奖励少而延迟。

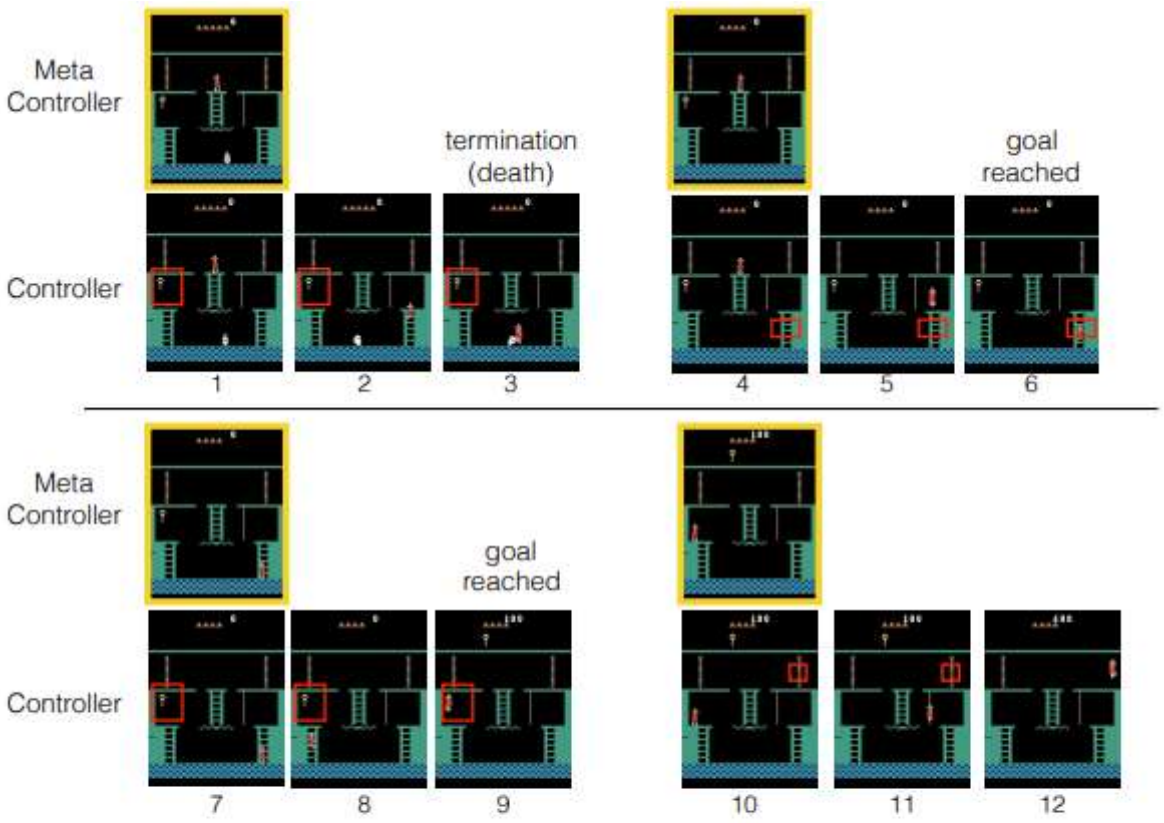

Figure 5: **Sample game play on Montezuma's Revenge:** The four quadrants are arranged in a temporal order (top-left, top-right, bottom-left and bottom-right). First, the meta-controller chooses key as the goal (illustrated in *red*). The controller then tries to satisfy this goal by taking a series of low level actions (only a subset shown) but fails due to colliding with the skull (the episode terminates here). The meta-controller then chooses the bottom-right ladder as the next goal and the controller terminates after reaching it. Subsequently, the meta-controller chooses the key and the top-right door and the controller is able to successfully achieve both these goals.

图5

1-3：agent始终只有一个，总控器选goal：钥匙，控制器根据当前位置和钥匙输出动作，下楼梯，往左走，但是碰到骷髅，挂了，critic判断终止。

4-6：总控器选下一个goal：右下梯子，控制器输出动作，agent实体走到右下梯子，critic判断goal达到。

7-9：总控器选下一个goal：钥匙，右上门，控制器输出动作，拿到钥匙，到达右上门，完成goal。

internal critic 以<entity 1, relation, entity 2>形式定义，例如：agent实体 到达 另一个实体 door，用两个实体的相对位置计算二进制reward。

结果：图4表明联合训练阶段的奖励进展——模型开始逐渐学习达到钥匙（+100）和开门（+300）从而获得每个回合大约+400的奖励。

agent首先学习“更简单”goals，比如到达右边的门或中间梯子，然后慢慢开始学习“更难”goals，比如钥匙和底下梯子，这些（更难的goals）可以提供获得更高奖励的途径。

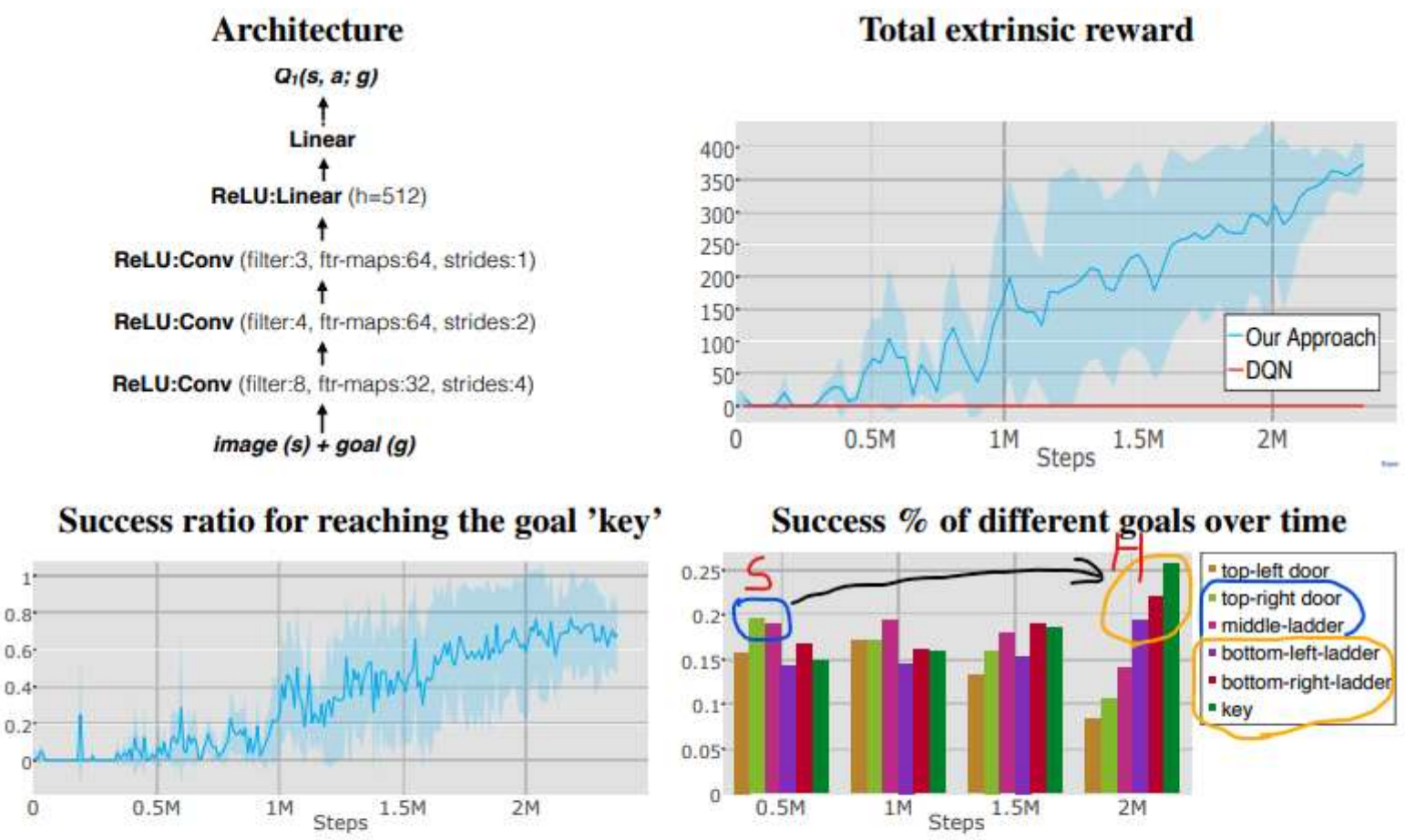

Figure 4: (**top-left**) *Architecture*: DQN architecture for the controller (Q_1). A similar architecture produces Q_2 for the meta-controller (without goal as input). (**top-right**) The joint training learns to consistently get high rewards. (**bottom-left**) *Goal success ratio:* The agent learns to choose the key more often as training proceeds and is successful at achieving it. (**bottom-right**) *Goal statistics:* During early phases of joint training, all goals are equally preferred due to high exploration but as training proceeds, the agent learns to select appropriate goals such as the key and bottom-left door.

训练结束阶段，我们可以看到：钥匙，底下左梯子，底下右梯子越来越经常地被选择。

edit (https://blogs.cuit.columbia.edu/zp2130/wp-admin/post.php?post=5207&action=edit)

Author: Z Pei (https://blogs.cuit.columbia.edu/zp2130/author/zp2130/) on April 12, 2019

Categories: AI (https://blogs.cuit.columbia.edu/zp2130/category/ai/), Algorithm (https://blogs.cuit.columbia.edu/zp2130/category/algorithm/), Hierarchical (https://blogs.cuit.columbia.edu/zp2130/category/hierarchical/), Machine Learning (https://blogs.cuit.columbia.edu/zp2130/category/machine-learning/), Reinforcement Learning (https://blogs.cuit.columbia.edu/zp2130/category/reinforcement-learning/), RL (https://blogs.cuit.columbia.edu/zp2130/category/rl/)

Tags: AI (https://blogs.cuit.columbia.edu/zp2130/tag/ai/), Algorithm (https://blogs.cuit.columbia.edu/zp2130/tag/algorithm/), Hierarchical RL (https://blogs.cuit.columbia.edu/zp2130/tag/hierarchical-rl/), Machine Learning (https://blogs.cuit.columbia.edu/zp2130/tag/machine-learning/), Reinforcement Learning (https://blogs.cuit.columbia.edu/zp2130/tag/reinforcement-learning/), RL (https://blogs.cuit.columbia.edu/zp2130/tag/rl/)

Other posts

Meta Learning Shared Hierarchies (https://blogs.cuit.columbia.edu/zp2130/meta_learning_shared_hierarchies/) «» Actor-Critic Algorithms for Hierarchical Markov Decision Processes (https://blogs.cuit.columbia.edu/zp2130/actor-critic_algorithms_for_hierarchical_markov_decision_processes/)

Last posts

- Symbolic Netlist to Innovus-friendly Netlist (https://blogs.cuit.columbia.edu/zp2130/symbolic_netlist_to_innovus-friendly_netlist/)
- Finite-Sample Convergence Rates for Q-Learning and Indirect Algorithms (https://blogs.cuit.columbia.edu/zp2130/finite-sample_convergence_rates_for_q-learning_and_indirect_algorithms/)
- Solving H-horizon, Stationary Markov Decision Problems In Time Proportional To Log(H) (https://blogs.cuit.columbia.edu/zp2130/paul_tseng_1990/)
- Randomized Linear Programming Solves the Discounted Markov Decision Problem In Nearly-Linear (Sometimes Sublinear) Run Time (https://blogs.cuit.columbia.edu/zp2130/randomized_linear_programming_solves_the_discounted_markov_decision_problem_in_nearly-linear_sometimes_sublinear_run_time/)
- KL Divergence (https://blogs.cuit.columbia.edu/zp2130/kl_divergence/)
- The Asymptotic Convergence-Rate of Q-learning (https://blogs.cuit.columbia.edu/zp2130/the_asymptotic_convergence-rate_of_q-learning/)
- Hierarchical Apprenticeship Learning, with Application to Quadruped Locomotion (https://blogs.cuit.columbia.edu/zp2130/hierarchical_apprenticeship_learning_with_application_to_quadruped_locomotion/)
- Policy Gradient Methods (https://blogs.cuit.columbia.edu/zp2130/policy_gradient_methods/)
- Actor-Critic Algorithms for Hierarchical Markov Decision Processes (https://blogs.cuit.columbia.edu/zp2130/actor-critic_algorithms_for_hierarchical_markov_decision_processes/)
- Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation (https://blogs.cuit.columbia.edu/zp2130/hierarchical_deep_reinforcement_learning_integrating_temporal_abstraction_and_intrinsic_motivation/)