Model-Free Episodic Control
&
Neural Episodic Control

presented by Jason TOKO

背景与动机

- DRL学习速度较慢：
  - 1.SGD优化一般需要较小的学习率；
  - 2.环境的奖励反馈稀疏；
  - 3.经验回放和目标网络使得奖励信息反向传播更慢。
- Episodic Control：一种memory-based的方法，利用已有的经验快速查找能产生高回报的动作。

背景与动机

- 大脑的学习机制：

基于记忆

海马体和相关内侧颞（niè）叶 ⟷ Episodic Control

基于逻辑、推理

前额叶皮质 ⟷ DRL

- 在不同场景，大脑学习、记忆和决策机制都有所不同。

Model-Free Episodic Control

- Model-Free Episodic Control建造了Q值表格来存储和回放经验
- 存储（更新）：

$$Q^{\mathrm{EC}}(s_t, a_t) \quad \leftarrow \begin{cases} R_t & \text{if } (s_t, a_t) \notin Q^{\mathrm{EC}}, \\ \max\left\{Q^{\mathrm{EC}}(s_t, a_t), R_t\right\} & \text{otherwise,} \end{cases}$$

- 回放（估计）：

$$\widehat{Q^{\mathrm{EC}}}(s, a) = \begin{cases} \frac{1}{k}\sum_{i=1}^{k} Q^{\mathrm{EC}}(s^{(i)}, a) & \text{if } (s, a) \notin Q^{\mathrm{EC}}, \\ Q^{\mathrm{EC}}(s, a) & \text{otherwise,} \end{cases}$$

Model-Free Episodic Control

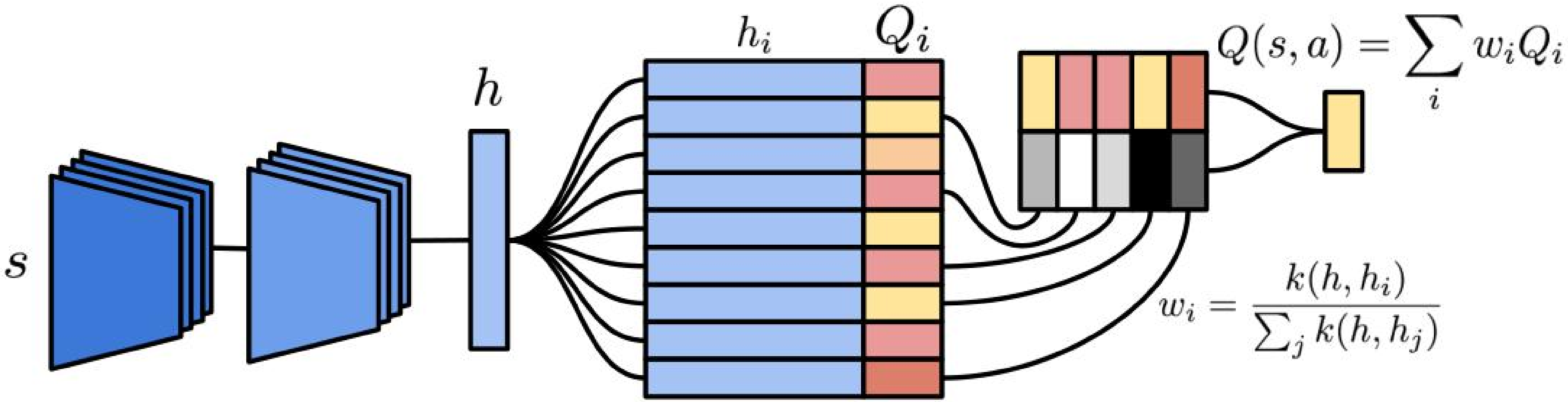
- **算法：**

Algorithm 1 Model-Free Episodic Control.

1: **for** each episode **do**
2: **for** $t = 1, 2, 3, \ldots, T$ **do**
3: Receive observation o_t from environment.
4: Let $s_t = \phi(o_t)$.
5: Estimate return for each action a via $\widehat{Q^{\mathrm{EC}}}(s,a) = \begin{cases} \frac{1}{k}\sum_{i=1}^{k} Q^{\mathrm{EC}}(s^{(i)}, a) & \text{if } (s,a) \notin Q^{\mathrm{EC}}, \\ Q^{\mathrm{EC}}(s,a) & \text{otherwise,} \end{cases}$
6: Let $a_t = \arg\max_a \widehat{Q^{\mathrm{EC}}}(s_t, a)$
7: Take action a_t, receive reward r_{t+1}
8: **end for**
9: **for** $t = T, T-1, \ldots, 1$ **do**
10: Update $Q^{\mathrm{EC}}(s_t, a_t)$ using R_t according to $Q^{\mathrm{EC}}(s_t, a_t) \leftarrow \begin{cases} R_t & \text{if } (s_t, a_t) \notin Q^{\mathrm{EC}}, \\ \max\left\{Q^{\mathrm{EC}}(s_t, a_t), R_t\right\} & \text{otherwise,} \end{cases}$
11: **end for**
12: **end for**

Neural Episodic Control

- Agent由三个部分组成：
  - 卷积网络：输入s，输出h
  - 可微神经字典(Differentiable Neural Dictionary，DND):输入h、a，输出w
  - 输出网络：输入w，输出Q(s,a)
- 结构图



Neural Episodic Control

- DND组成：每一个动作$a \in \mathcal{A}$各对应一个记忆模块$M_a = (K_a, V_a)$
  K_a为关键字h_i的集合，V_a为值v_i的集合
- DND查找
  - 1、通过$k(x, y)$计算关键字h与字典关键字h_i的kernel值，计算权值

$$w_i = k(h, h_i) / \sum_j k(h, h_j),$$

  - 2、加权求和

$$o = \sum_i w_i v_i,$$

Neural Episodic Control

- DND更新
  - 使用N-step Q-value作为DND更新目标

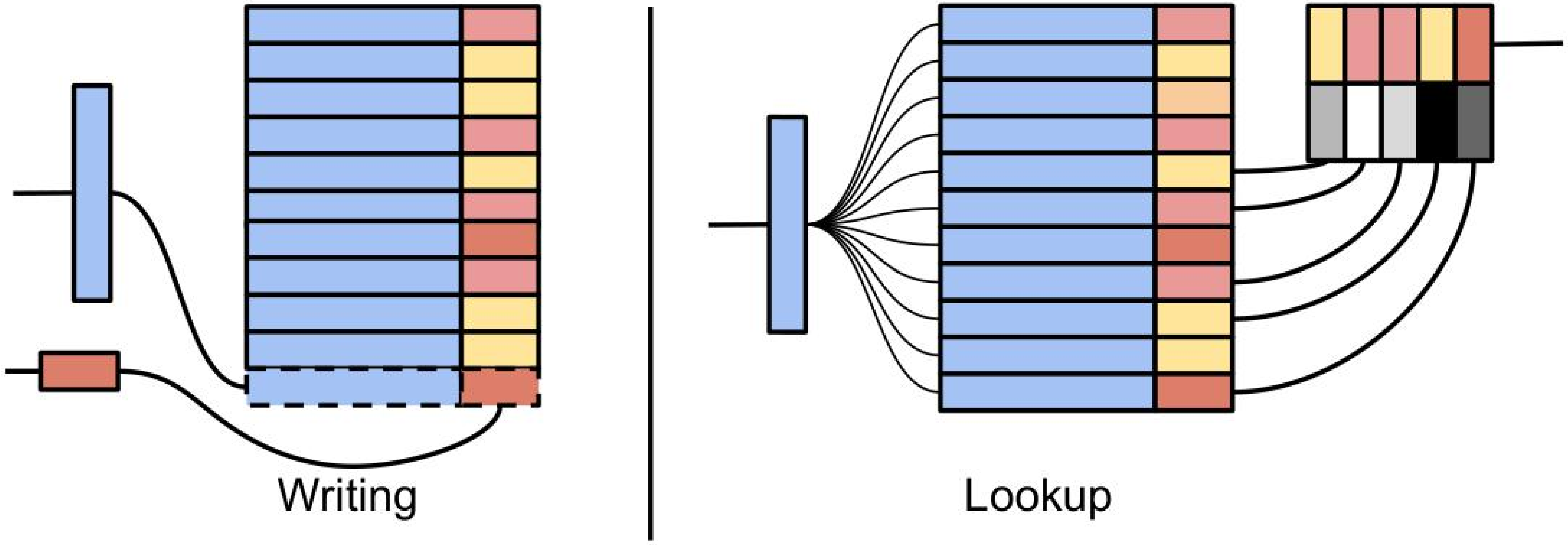
$$Q^{(N)}(s_t, a) = \sum_{j=0}^{N-1} \gamma^j r_{t+j} + \gamma^N \max_{a'} Q(s_{t+N}, a')$$

  - 若关键字h不存在于字典，则直接加入DND；
  - 若关键字h已存在于字典，则使用Q-learning方法更新:

$$Q_i \leftarrow Q_i + \alpha(Q^{(N)}(s, a) - Q_i)$$

Neural Episodic Control

- DND查找与更新



Neural Episodic Control

- 整个Agent训练：
  - 从replay buffer中随机采样minibatch(s_t, a_t, R_t) 。其中 $R_t = Q^N(s_t, a_t)$
  - 损失函数：

$$L = \frac{1}{M} \sum_{(s_t, a_t, R_t)} [R_t - Q(s_t, a_t)]^2$$

Neural Episodic Control

- 算法：

Algorithm 1 Neural Episodic Control

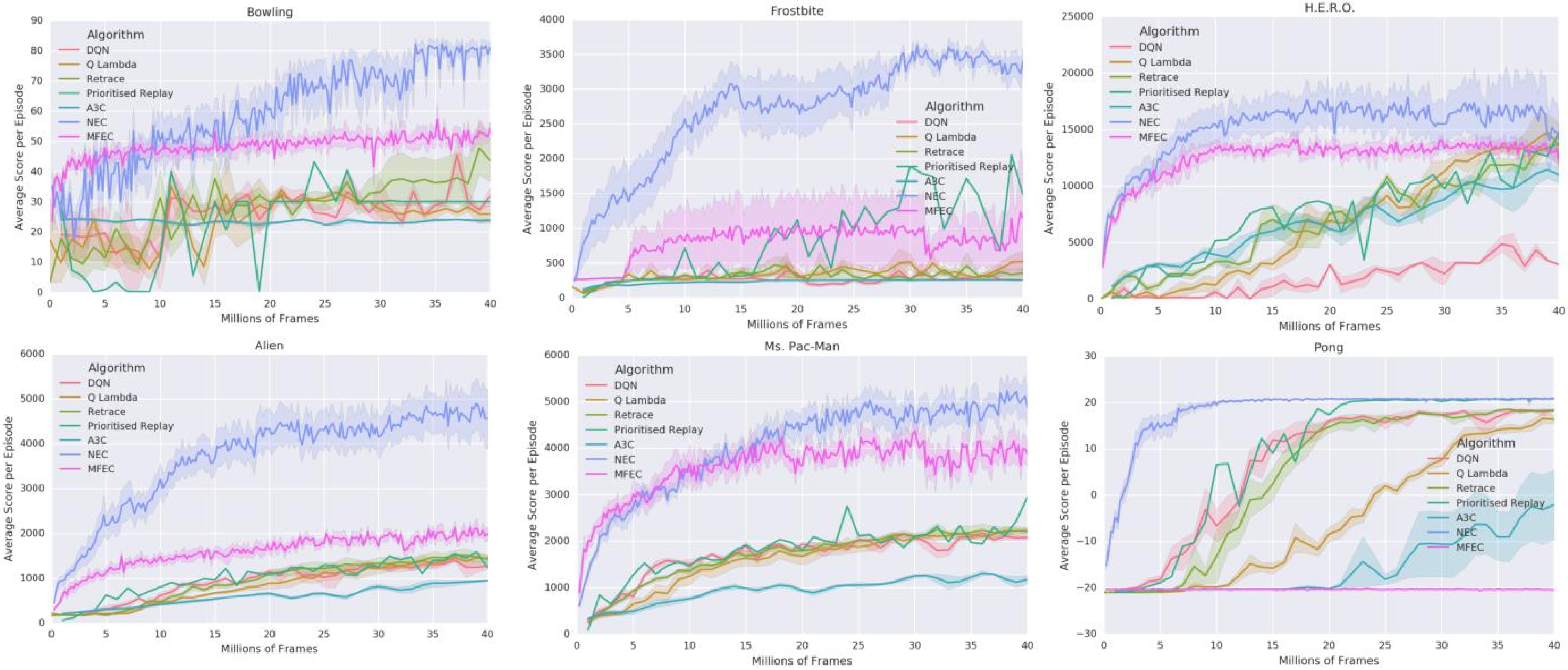
$\mathcal{D}$: replay memory.
M_a: a DND for each action a.
N: horizon for N-step Q estimate.
for each episode **do**
  for $t = 1, 2, \ldots, T$ **do**
    Receive observation s_t from environment with embedding h.
    Estimate $Q(s_t, a)$ for each action a via (1) from M_a
    $a_t \leftarrow \epsilon$-greedy policy based on $Q(s_t, a)$
    Take action a_t, receive reward r_{t+1}
    Append $(h, Q^{(N)}(s_t, a_t))$ to M_{a_t}.
    Append $(s_t, a_t, Q^{(N)}(s_t, a_t))$ to $\mathcal{D}$.
    Train on a random minibatch from $\mathcal{D}$.
  end for
end for

至少N步以后才能加入

More……

- Model-Free Episodic Control和Neural Episodic Control皆使用了字典来实现Q值查找和更新，考虑内存限制以及效率，有以下措施：
  - 1、限制表格大小，溢出时替换最近访问次数最少的状态；
  - 2、使用K-邻近状态来更新而非整个字典，并使用KD树实现查找

实验


Bowling
Frostbite
H.E.R.O.
Alien
Ms. Pac-Man
Pong
Algorithm
DQN
Q Lambda
Retrace
Prioritised Replay
A3C
NEC
MFEC
Average Score per Episode
Millions of Frames

Episodic Control与DRL对比

- 优点：解决DRL存在的三个问题，通过不断存储和再现经验，实现快速学习。
- 缺点：Episodic Control 通用性较差，更适用于在exploitation比exploration重要且相对来说噪音比较少的环境。