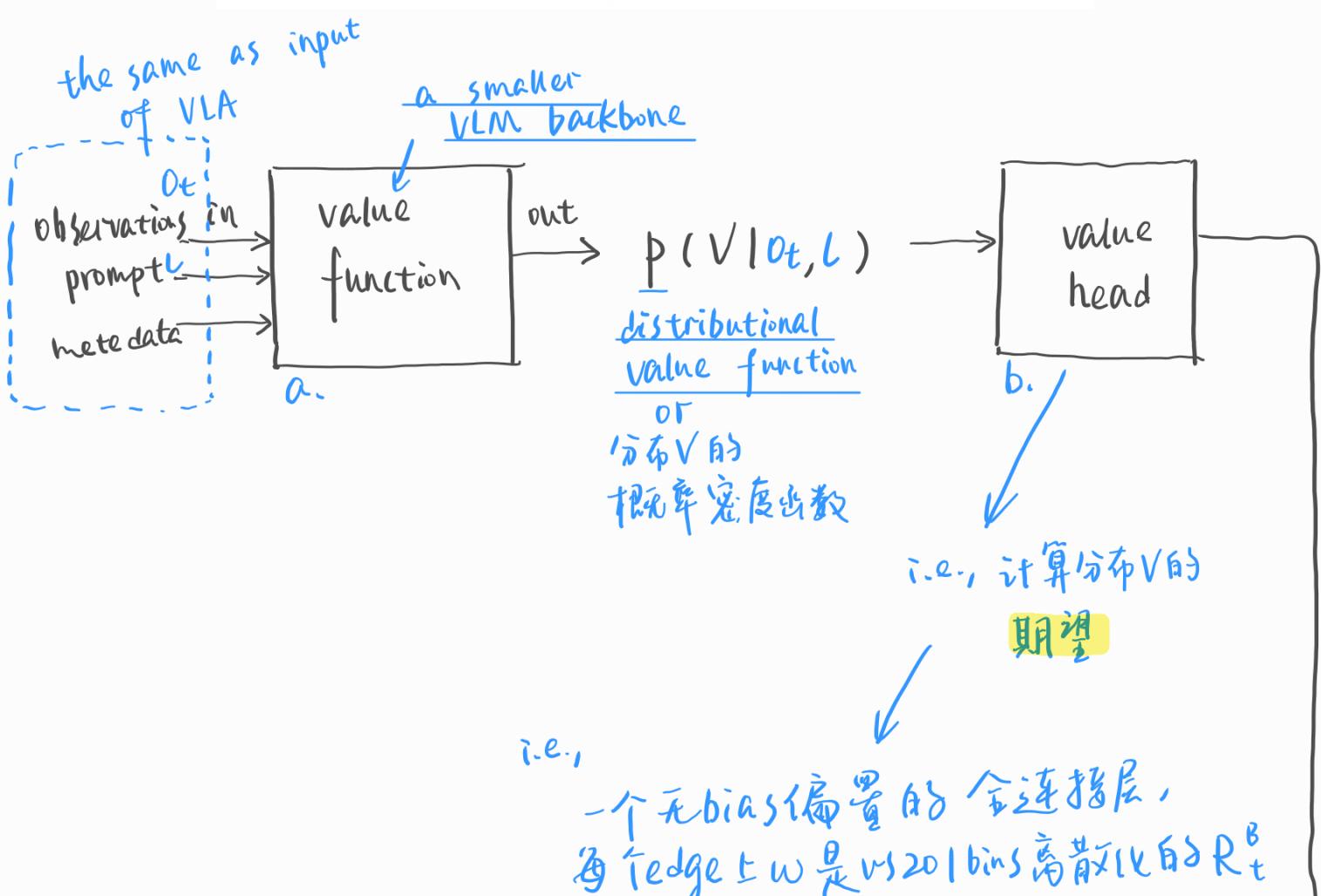
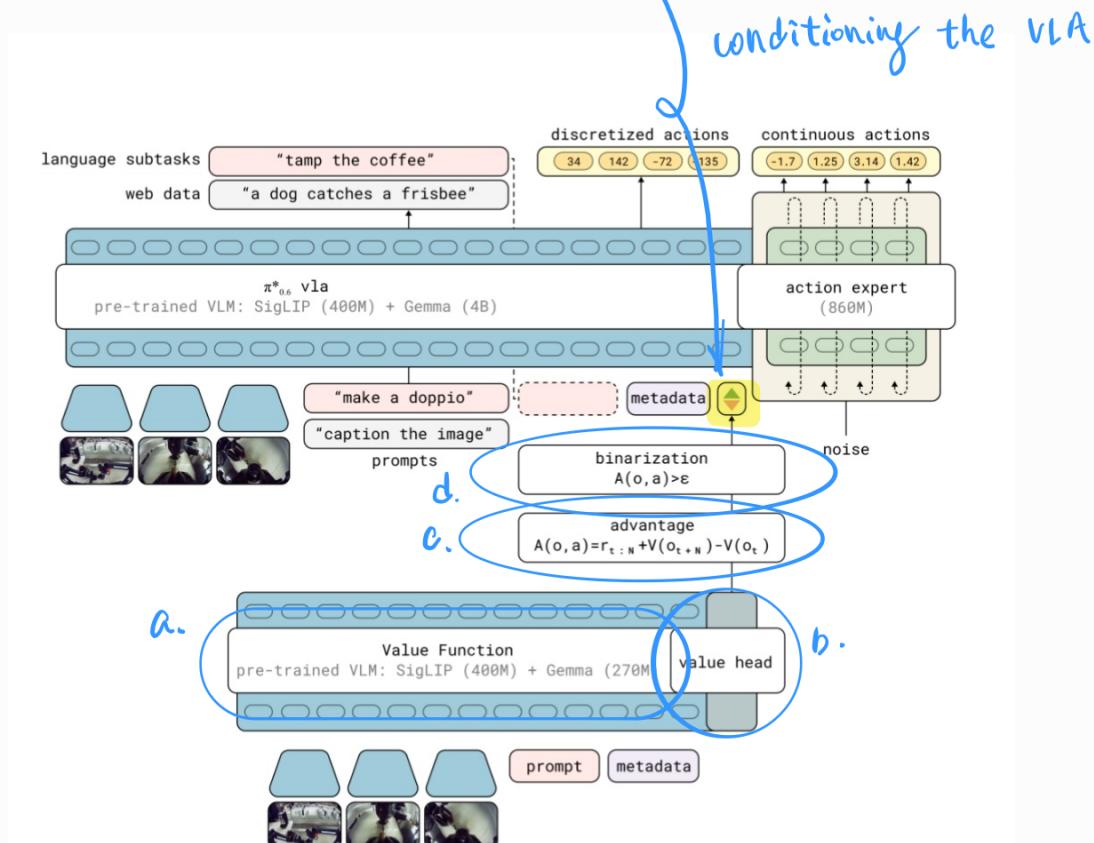


RECAP - RL with Experience and Corrections via Advantage-conditioned Policies

KEY FOCUS

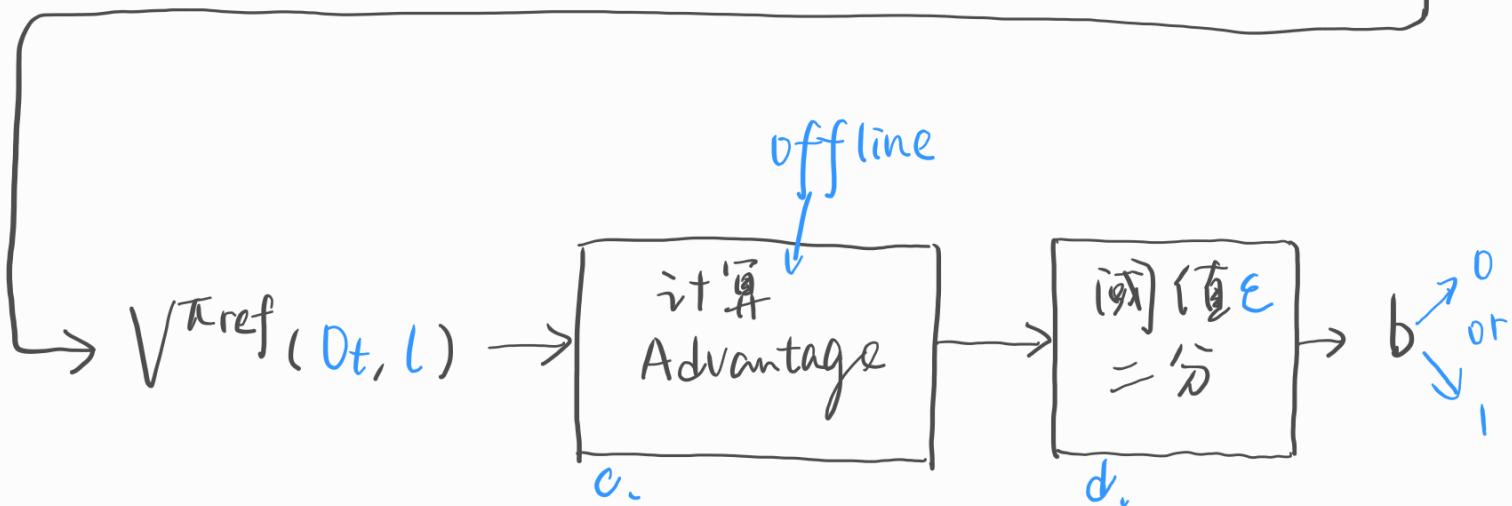
1. How to get the binarized indicator



对于某 t 时刻的某一由当前 π_{ref}
输出的轨迹 τ

$$P(R_t^{B_1} | o_{t,l}) \xrightarrow{R_t^{B_1}} \\ P(R_t^{B_2} | o_{t,l}) \xrightarrow{R_t^{B_2}} \dots \xrightarrow{R_t^{B_3}} \xrightarrow{+} V^{\pi_{ref}}$$

$$P(R_t^{B_{201}} | o_{t,l}) \xrightarrow{R_t^{B_{201}}}$$



2. 如何训练 Value function 即 $P(V|o_t, l)$?

1) Receipt

step 1 用当前 policy π_{ref}
rollout 得到多段轨迹的集合 D



step 2 对集合 D Monte Carlo 抽样，
通过交叉熵 loss 更新 value function

$$\min_{\phi} \mathbb{E}_{\tau \in D} \left[\sum_{o_t \in \tau} H(R_t^B(\tau), p_{\phi}(V|o_t, l)) \right]. \quad (1)$$

Q: Equation 1 表达了什么?

传统的交叉熵公式 $H(P, Q)$ 中两个变量均为分布，
但 Equation 1 中 $R_t^B(\tau)$ 是一个 constant，
 $P(V|o_t, \ell)$ 是概率密度函数

A (by Gemini):

- 应该将 $R_t^B(\tau)$ 视为一个 One-Hot Encoding (像脉冲信号) 分布，即
- $$P(b) = \begin{cases} 1, & b = R_t^B(\tau) \\ 0, & b \neq R_t^B(\tau) \end{cases}$$
- 因此，展开 Equation (1) 得到，

Step 1: Substitute the definition of H We treat $R_t^B(\tau)$ as the target distribution P and p_ϕ as the predicted distribution Q :

$$\min_{\phi} \mathbb{E}_{\tau \in \mathcal{D}} \left[\sum_{o_t \in \tau} \left(- \sum_{b=0}^B P(b) \log p_\phi(V = b | o_t, \ell) \right) \right]$$

Step 2: Apply the One-Hot logic Since $P(b) = 1$ only when $b = R_t^B(\tau)$ and is 0 otherwise, the inner summation collapses. All terms where $b \neq R_t^B(\tau)$ become zero:

$$\min_{\phi} \mathbb{E}_{\tau \in \mathcal{D}} \left[\sum_{o_t \in \tau} (-1 \cdot \log p_\phi(V = R_t^B(\tau) | o_t, \ell)) \right]$$

Step 3: Final Expanded Form The simplified objective function, which is what is actually implemented in code (often called the **Negative Log-Likelihood**), is:



$$\min_{\phi} \mathbb{E}_{\tau \in \mathcal{D}} \left[\sum_{o_t \in \tau} -\log p_\phi(V = R_t^B(\tau) | o_t, \ell) \right]$$

Note:

1. 随机变量 $V \in [\min R_t^B(\tau), \max R_t^B(\tau)]$,

与当前轨迹有关 (不同 R_t 的离散结果不同)

2. 分布 V 的随参密度函数用 VLM 建模

附,

Standard Definition of Cross-Entropy

For two discrete probability distributions P (the ground truth) and Q (the prediction) over the same set of outcomes \mathcal{X} , the Cross-Entropy $H(P, Q)$ is defined as:

$$H(P, Q) = - \sum_{x \in \mathcal{X}} P(x) \log Q(x)$$

2) On-policy Learning

· value function 只会被 π^{ref} 的轨迹更新，而无法使用到 human 数据 or imitation learning 等方法

While this on-policy estimator is less optimal than a more classic off-policy Q-function estimator, we found it to be simple and highly reliable, while still allowing for substantial improvement over imitation learning. Our method could be extended to accommodate off-policy estimators in future work.

interesting

Q: 如何规避过拟合？

传统 on-policy 方法易陷入过拟合，
需要引入 exploration, e.g. greedy $\rightarrow \epsilon$ -greedy

A: Co-train with web data

The value function takes as input the same language inputs as the $\pi_{0.6}^*$ VLA, and uses the same architecture design, with a smaller 670M parameter VLM backbone that is also initialized from Gemma 3 (see Figure 3). To prevent overfitting, we also co-train the value function on a small mixture of multi-modal web data. Figure 4 show visualizations of the value function on some examples of successful and failure episodes, with additional visualizations in Figure 13 in Appendix B.