PPO (Proximal Policy Optimization)

Kieu Giang Bien

Ngày 2 tháng 9 năm 2025

Tổng quan

- **PPO** (Proximal Policy Optimization) là một họ các phương pháp policy-gradient hiện đại, tối ưu sự ổn định và hiệu quả.
- Ý tưởng chính: giới hạn lượng thay đổi policy trong mỗi bước cập nhật để tránh phá vỡ policy hiện tại (trust-region-like).
- **PPO + CNN**: áp dụng PPO với policy/value network có extractor dạng CNN — cần cho các tác vụ có quan sát là ảnh (Atari, Robotics từ camera, v.v).

PPO: Clipped Surrogate Objective

- Đặt: π_θ là policy tham số hoá bởi θ. Tỉ số xác suất:

$$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}.$$

- Hàm mục tiêu **clipped**:

$$L^{\text{CLIP}}(\theta) = \mathbb{E}_t\Big[\min\left(r_t(\theta)\hat{A}_t,\ \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\Big].$$

- Ý nghĩa: nếu $r_t(\theta)$ thay đổi vượt quá ϵ, ta dùng giá trị đã bị cắt để tránh cập nhật quá lớn.

Hàm mất tổng hợp — Value Entropy

- Value loss (mean-squared):

$$L^V(\theta) = \mathbb{E}_t \left[\left(V_\theta(s_t) - V_t^{\text{target}} \right)^2 \right].$$

- Entropy bonus (khuyến khích khám phá):

$$S[\pi_\theta](s_t) = -\mathbb{E}_{a \sim \pi_\theta(\cdot|s_t)}[\log \pi_\theta(a|s_t)].$$

- Tổng loss (minimize):

$$L(\theta) = -L^{\text{CLIP}}(\theta) + c_1 L^V(\theta) - c_2 \mathbb{E}_t \left[S[\pi_\theta](s_t) \right].$$

Trong đó c_1, c_2 là hệ số cân bằng.

Ước lượng lợi thế: GAE (Generalized Advantage Estimation)

- Độ lợi thế (advantage) ước lượng bằng GAE:

$$\delta_t = r_t + \gamma V(s_{t+1}) - V(s_t)$$

$$\hat{A}_t = \sum_{l=0}^{\infty} (\gamma\lambda)^l \delta_{t+l}$$

- λ điều chỉnh bias-variance: $\lambda = 0 \Rightarrow$ giống TD(0); $\lambda = 1 \Rightarrow$ Monte-Carlo.
- GAE giúp ổn định và giảm phương sai cho gradient policy.

Tại sao dùng **clipping** thay vì TRPO?

- TRPO tối ưu với ràng buộc KL nhưng phức tạp (requires conjugate gradient).
- PPO (clipped) đơn giản, dễ triển khai, vẫn giữ được tính ổn định bằng cách cắt $r_t(\theta)$.
- Hiệu năng thực nghiệm cho thấy PPO cân bằng tốt giữa ổn định và hiệu quả.

Pseudocode: PPO (mini-batch, multiple epochs)

- Thu thập một batch rollout $(s_t, a_t, r_t, \log \pi_{\text{old}}(a_t|s_t), V_t)$.
- Tính $\hat{A}_t$ bằng GAE và V_t^{target}.
- Với nhiều epoch:
  1. Shuffle batch, chia thành mini-batches.
  2. Tính $r_t(\theta)$, L^{CLIP}, L^V, entropy.
  3. Cập nhật θ bằng gradient descent trên tổng loss: $L(\theta)$.

PPO + CNN: Kiến trúc chi tiết

- Khi s_t là ảnh (stacked frames), ta dùng CNN để trích xuất đặc trưng $\phi(s_t)$.
- Tiếp theo chia làm 2 head:
  - **Policy head**: dự đoán phân phối $\pi_\theta(a|s)$ (softmax cho discrete, gaussian cho continuous).
  - **Value head**: ước lượng $V_\theta(s)$.
- Cấu trúc tổng quát:

$$s_t \xrightarrow{\text{CNN}} \phi(s_t) \xrightarrow{\text{MLP}} \begin{cases} \text{policy logits} \\ \text{value} \end{cases}$$

Ví dụ module CNN (kiến trúc mẫu)

- Một CNN điển hình cho Atari:
  - Conv(32, 8x8, stride=4) + ReLU
  - Conv(64, 4x4, stride=2) + ReLU
  - Conv(64, 3x3, stride=1) + ReLU
  - Flatten $\rightarrow$ FC(512) + ReLU
- Sau đó tách 2 head: policy logits và value (1 node).

Thu thập kinh nghiệm và cập nhật

- Thu thập T bước hoặc nhiều episodes cho mỗi iteration.
- Tính $\hat{A}_t$ (GAE) và V_t^{target}.
- Thực hiện nhiều epoch tối ưu hóa trên batch đã thu thập, shuffle chia mini-batches.
- Giám sát: reward trung bình, KL divergence, entropy, loss.

Hyperparameters phổ biến cho PPO

- learning rate: $1e-4$ – $3e-4$
- discount γ: 0.99 – 0.999
- GAE λ: 0.95 – 0.98
- clip ϵ: 0.1 – 0.3
- epochs per update: 3 – 10
- mini-batch size: 64 – 1024 (tùy batch)
- entropy coeff c_2: 0.0 – 0.01, value coeff c_1: 0.5 – 1.0

Lưu ý khi huấn luyện PPO + CNN

- Chuẩn hoá / scale reward khi cần.
- Clip gradient, sử dụng Adam hoặc RMSProp.
- Khi dùng image input: tiền xử lý (grayscale, resize, stack frames).
- Theo dõi KL giữa policy mới và cũ; nếu tăng quá nhanh có thể giảm LR hoặc tăng số epoch.
- Regularize: weight decay nhẹ, gradient norm clip.

Tổng kết

- PPO là phương pháp dễ triển khai, ổn định cho RL policy-gradient.
- Clipped surrogate objective là trái tim của PPO — ngăn cập nhật policy quá lớn.
- Với quan sát dạng ảnh, tích hợp CNN làm feature extractor rất hiệu quả — tách rõ policy head và value head.
- Thực nghiệm: cần tinh chỉnh hyperparameters (clip, epochs, batch size, λ).

Tài liệu tham khảo

- Schulman et al., "Proximal Policy Optimization Algorithms"(2017).
- Schulman et al., "High-Dimensional Continuous Control Using Generalized Advantage Estimation"(2015).
- OpenAI Baselines / Stable-Baselines3 implementations.