UNREAL における補助タスクの適応的選択

EP15009 板谷英典　　指導教授：山下隆義

1. はじめに

強化学習とは，数値化された報酬を最大とするために，未知の環境において，エージェントが何をすべきか学習する問題である．深層強化学習法の一つである UNREAL [1] は異なる複数の補助タスクを学習時に導入することで，ゲームタスクにおいて高いスコアを達成している．しかし，UNREAL で用いられる全ての補助タスクが，あらゆる環境において必ずしも有効であるとは限らない．そこで，本研究では補助タスクを問題に合わせ適応的に選択することで，効率的に学習する手法を提案する．

2. UNREAL

UNREAL は，非同期的な学習法である Asynchronous Advantage Actor-Critic (A3C) [2] をベースに，教師なし学習の異なる 3 つの補助タスクをメインタスクと並列に実行する手法である．UNREAL における損失関数 L_{UNREAL} を式 (1) に示す．

$$L_{\mathrm{UNREAL}} = L_{\mathrm{A3C}} + L_{\mathrm{VR}} + \sum_c L_Q^{(c)} + L_{\mathrm{RP}} \quad (1)$$

第 1 項の L_{A3C} はメインタスクである A3C の損失関数，第 2 項から 4 項は各補助タスクの損失関数であり，L_{VR} は Value Function Replay，$\sum_c L_Q^{(c)}$ は Pixel Control，L_{RP} は Reward Prediction である．Value Function Replay は過去の経験をシャッフルし，状態価値関数 $V(s)$ を学習するタスクである．Pixel Control は画像の画素が大きく変化する行動を学習するタスクである．Reward Prediction は報酬を獲得した経験を優先して学習し，未来の報酬を予測するタスクである．

3. 提案手法

UNREAL の各補助タスクは，環境によってその有効性が異なるため，メインタスクの学習を妨げるという問題がある．本研究では，環境に合わせてどの補助タスクを用いるかを適応的に選択するタスク Auxiliary Selection を提案する．

3.1. Auxiliary Selection

図 1 に Auxiliary Selection を導入した UNREAL のネットワーク構成を示す．Auxiliary Selection には，Replay Buffer 内に格納された画像を入力し，状態価値関数 $V_{\mathrm{AS}}(s)$ と方策 π_{AS} を出力する．方策 π_{AS} は各補助タスクを用いるかどうかを表す値である．各補助タスクに対する重みを $C_{\mathrm{PC}} = \{0, 1\}$，$C_{\mathrm{VR}} = \{0, 1\}$，$C_{\mathrm{RP}} = \{0, 1\}$ とするとき，$\pi_{\mathrm{AS}} = (C_{\mathrm{PC}}, C_{\mathrm{VR}}, C_{\mathrm{RP}})$ と表す．このとき，Auxiliary Selection のネットワークは A3C のネットワークを共有しない．このように，環境に合わせて補助タスクを適応的に選択することで，補助タスクを設計する際の効率化を図る．

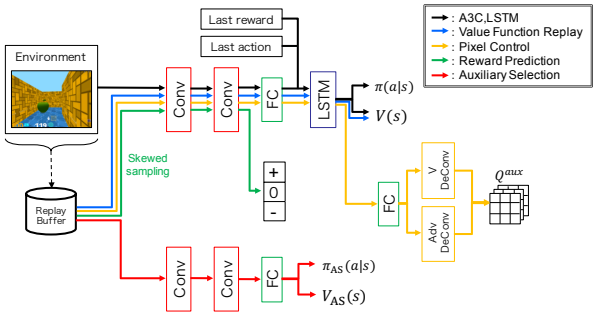

図 1：提案手法のネットワーク構成

3.2. 損失関数

従来の UNREAL の損失関数である式 (1) をもとに，提案手法の損失関数 L_{proposed} を式 (2) のように定義する．

$$L_{\mathrm{proposed}} = L_{\mathrm{A3C}} + C_{\mathrm{VR}} L_{\mathrm{VR}} + C_{\mathrm{PC}} \sum_c L_Q^{(c)} + C_{\mathrm{RP}} L_{\mathrm{RP}} \quad (2)$$

提案手法では，Auxiliary Selection から獲得する C_{VR}，C_{PC}，C_{RP} と各補助タスクの損失関数の積を取ることで，最適な補助タスクのみを用いた学習を実現する．

4. 評価実験

提案手法の有効性を評価実験により検証する．

4.1. 実験概要

評価データとして，Deep Mind Lab [3] の nav_maze_static_01 (maze)，seekavoid_arena_01 (seekavoid)，lt_horseshoe_color (horseshoe) の 3 種類のゲームを用いる．本実験では，全補助タスクを用いた場合 (UNREAL)，Pixel Control のみの場合 (PC)，Value Function Replay のみの場合 (VR)，Reward Prediction のみの場合 (RP)，提案手法 (proposed) を比較する．maze 及び seekavoid では 5.0×10^7 ステップ，horseshoe では 1.0×10^8 ステップまで学習を行い，ステップ毎のスコアを比較する．

4.2. 実験結果

各ゲームにおけるスコアの推移を図 2 に示す．従来の UNREAL および特定の補助タスクを用いた場合を比較すると，maze では UNREAL 及び PC，seekavoid では VR，horseshoe では UNREAL が最適な補助タスクの選択であることが分かる．一方，提案手法は maze 及び horseshoe において，UNREAL と同等のスコアを達成した．また，seekavoid では，最適な補助タスクを選択することで，UNREAL のスコアを上回り，最も高いスコアを獲得した．

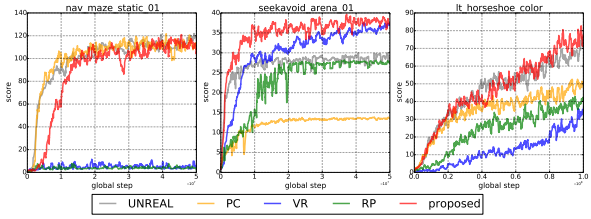

図 2：各ゲームにおける学習ステップ毎のスコア

4.3. 考察

各ゲームの 1 エピソードにおける補助タスクの選択回数を表 1 に示す．ここで，選択回数とは 50 エピソード間の平均の選択回数であり，括弧内は 1 エピソード内で選択する割合を表す．seekavoid では最適な補助タスクである VR，horseshoe では PC と RP を安定して選択し，maze では全ての補助タスクが同等に選択されている．maze は UNREAL と同様に全ての補助タスクを選択するため，最適な補助タスクである UNREAL と同等のスコアを獲得したと考えられる．したがって，UNREAL に Auxiliary Selection を導入することで，環境に合わせた補助タスクを選択でき，効率的な学習を実現していると言える．

表 1：1 エピソードにおける補助タスクの選択回数

環境 \ 補助タスク	PC	VR	RP
maze	435.4(48.3%)	487.8(54.1%)	369.0(41.0%)
seekavoid	0.3(0.1%)	300.0(100.0%)	0.0(0.0%)
horseshoe	8545.1(94.9%)	14.1(0.1%)	8998.2(99.9%)

5. おわりに

本研究では，学習に用いる補助タスクを適応的に選択する Auxiliary Selection を提案した．DeepMind Lab を用いた実験により，効率的に学習できることを示した．今後は，新たな補助タスクを導入し，多様な補助タスクでの提案手法の有効性などが挙げられる．

参考文献

[1] M. Jaderberg, *et al.*, “Reinforcement Learning with Unsupervised Auxiliary Tasks”, ICLR, 2017.

[2] V. Mnih, *et al.*, “Asynchronous methods for deep reinforcement learning”, ICML, 2016.

[3] C. Beattie, *et al.*, “DeepMind Lab”, arXiv preprint, arXiv:1612.03801, 2016.