熟練した他者からのメッセージを通じた自己モデルの獲得の検討

Investigation of a Self-model based on messages from another skilled agent

福地庸介 [1] * 大澤正彦 [1] 岨野太一 [1] 山川宏 [2] 今井 倫太 [1]
Yosuke Fukuchi[1] Masahiko Osawa[1] Taichi Sono[1] Hiroshi Yamakawa[2] Michita Imai[1]

[1] 慶應義塾大学
[1] Keio University
[2] 株式会社ドワンゴ　ドワンゴ人工知能研究所
[2] DWANGO Co., ltd Dwango Artificial Intelligence Laboratotry

Abstract: 機械学習によって獲得された制御モデルの行動に対し，自ら解釈を与える自己モデルを生成させることで，人間に理解される深層学習エージェントを構築することを目的とする．具体的には，既に熟練した制御モデルを持つエージェントと未熟エージェントの二者を設定し，未熟エージェントは熟練エージェントのメッセージをもとに自己モデルを構成する．また熟練エージェントが適切な自己モデルを生成できたとき，その自己解釈は優れた戦略になると仮定し，メッセージが未熟エージェントの学習に寄与するかで獲得した自己モデルを評価する．

1 はじめに

深層学習によって獲得されるモデルは大量の数値パラメータで表現され，その内部状態はブラックボックスとなってしまう．そのため深層学習モデルによる意思決定のプロセスを人間が理解するのは難しい．制御モデルに異常が起こった場合も，人間はモデルへ直接入出力される情報のみから異常の理由を推定しなければならず，それは困難である．

Whiten は人間による対象理解の戦略として，対象が何らかの機能やアルゴリズムをもとに行動していると仮定する behavior-reading と，対象が何らかの意図を持って行動していると仮定する mind-reading の 2 つを示した [1]．深層学習エージェントはその複雑な内部状態をもとに行動決定するため，行動を一つのアルゴリズムで説明することは難しく，人間はエージェントに対し mind-reading を行って，意図という観点から行動を理解する必要があると考えられる．

一方エージェントの内部モデルの理解という問題は，ヒューマンエージェントインタラクションの分野でも他者認知という観点から扱われてきた．人間は他者の置かれている環境状態や振る舞いから，直接観測できない他者の内部状態を推定し，推定した内部状態に基づいて他者の意図を理解している．本稿では，この他者が置かれている環境の状態や振る舞いをもとに他者の意図を解釈する認知モデルを他者モデルと呼ぶ．また，この他者モデルの他者を自己に置き換えた自己モデルとする．つまり自己モデルを，自己が置かれている環境の状態や自身の振る舞いをもとに自身の意図を解釈するモデルとする．

他者認知に対する代表的な立場として，人間が他者モデルについてあらかじめ持つ理論的知識に基づく推論によって他者を理解するという理論説と，自己モデルを他者に投影することで他者を理解するというシミュレーション説があげられる [2]．しかしいずれにしても，自己と他者は，他者モデルを構築するために最低限必要な，基礎となるような枠組みを，前者は理論として，後者は自己の投影として共有していることが前提となる．そのため人間に理解される深層学習エージェントと人との間にも，この基礎が共有されている必要があると考える．

そこで本稿では，人間が与えた戦略を基礎に自己モデルを構築するシステムを検討する．人間が与えた戦略を基礎とさせることで，構築された自己モデルが出力する行動の解釈が人間の認識に寄り添ったものになり，人間によるエージェントの理解を助けるような情報が抽出される，ということを期待している．

*連絡先：慶應義塾大学理工学部情報工学科
〒 223–0061 神奈川県横浜市港北区日吉 3–14–1
E-mail: fukuchi@ailab.ics.keio.ac.jp

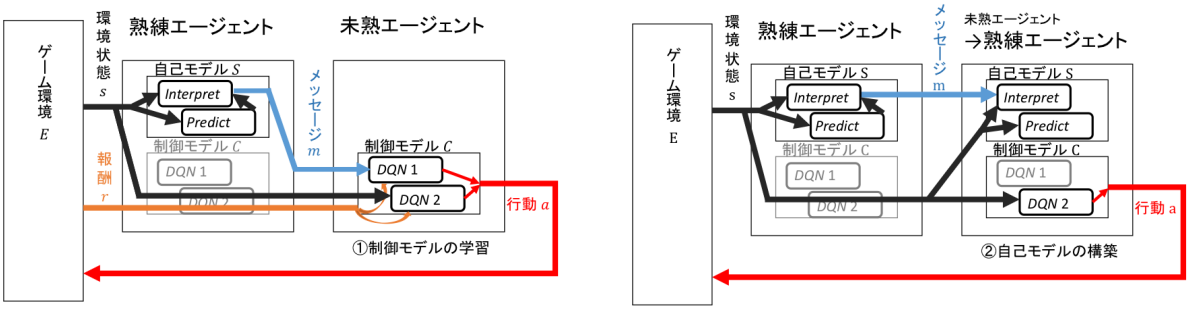

図 1: 未熟エージェントによる制御モデルの学習 (左) と，自己モデルの構築 (右)

2　システム概要

2.1　提案システムの構成

図 1 にシステムの構成図を示す．本稿で考えるのは，制御モデルと自己モデルの 2 つのモデルを持つエージェントである．制御モデル C は Deep Q Network アルゴリズム [3] を用いた深層強化学習器を 2 つ持つ．一方の学習器は直前 3 フレーム分の環境の状態 s を，他方は熟練エージェントからのメッセージ m を入力とし，エージェントの取るべき行動 α を出力する．制御モデル C は，2 つの DQN からの出力と，2.3 章で説明する方策選択に関与する変数 ϵ をもとに，エージェントが取る行動 a を選択する．

$$DQN_1(m_{t-2}, m_{t-1}, m_t) = \alpha_{1t}$$

$$DQN_2(a_{t-2}, a_{t-1}, a_t) = \alpha_{2t}$$

$$C(\epsilon, \alpha_{1t}, \alpha_{2t}) = a_t$$

環境 E は，エージェントの行動 a を受けて次の状態へ遷移し，エージェントに対し報酬 r を与える．

$$E_t(a_t) = s_{t+1}, r_t$$

一方自己モデル S は，自身の行動による環境の変化を予測する関数 $Predict$ と，環境の変化から過去の行動の意図を解釈し，ビット列のメッセージ m として表現する関数 $Interpret$ からなる．

$$Predict(s_{t-2}, s_{t-1}, s_t) = s^*_{t+1}$$

$$Interpret(s_{t-2}, s_{t-1}, s_t) = m_{t-2}$$

自己モデル S は，時刻 t における自身の制御モデルの行動が，いかなる意図に基づくかを表すメッセージ m^*_t を出力する．

$$S(s_{t-2}, s_{t-1}, s_t) = Interpret(s_t, s^*_{t+1}, s^*_{t+2}) = m^*_t$$

制御モデルの真の意図と整合性のある自己モデルが獲得できている時，m^*_t は自身が持つ制御モデルの戦略を説明していると言える．

システムの流れは大きく二つに分けられる．第一段階は，未熟エージェントが，熟練エージェントからのメッセージを利用しながら制御モデルを学習する段階，そして第二段階は，未熟エージェントが熟練エージェントからのメッセージをもとに，自己モデルを構築する段階である．

学習済み制御モデルと整合性のある自己モデルが獲得できていれば，獲得した自己モデルのメッセージは，他の未熟エージェントの第一段階の学習を加速させるような情報を持っていると考えられる．そのため第二段階で新たに自己モデルを獲得したエージェントは，新たな熟練エージェントとして別の未熟エージェントにメッセージを送り，この時の未熟エージェントによる制御モデルの学習の進度によって，新しい熟練エージェントが獲得した自己モデルを評価する．

2.2　制御モデルの学習

未熟エージェントがメッセージを利用して学習を有利に進める機構を構築するため，エージェントはメッセージから行動を選択する DQN 学習器を持っている．メッセージは環境状態よりも情報が削減されていて単純である．そのためこのメッセージを扱う学習器は，完璧な制御には向かないが，環境状態を扱う学習器に比べて単純なモデルで済み，早く学習すると考えられる．これを踏まえ今回は ϵ-greedy 法を応用し，方策選択を図 2 で表されるような確率で決定することにした．これにより，探索のためのランダム行動の一部がメッセージによる学習器が出力する行動で置き換わることになる．制御モデルは，ランダム探索のみの場合に比べて有利な経験を得やすくなるため，より少ない施行数で学習が進むと考えられる．

2.3　自己モデルの構築

制御モデルを学習したエージェントは，他者からのメッセージをもとにして，自己の意図を表現するモデ

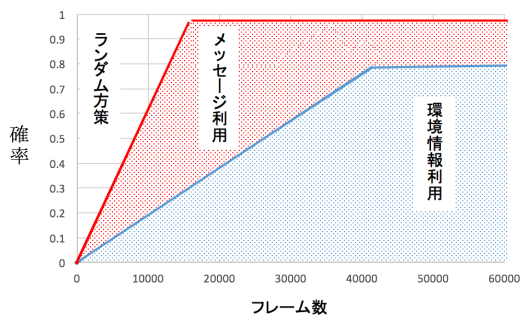

図 2: フレーム数に応じた行動選択の確率．各領域の縦幅が，制御モデルの学習の際にそれぞれの方策を採用する確率を示す．

ルを構築する．このとき，行動がもたらした環境の変化に着目して，行動を解釈する．

具体的には，はじめに未熟エージェントの学習済みの制御モデルを使い，環境状態と，その際得られる熟練エージェントからのメッセージの組 (s_t, m'_t) を収集する．自己モデルの獲得には，ここで得られた情報を用いる．

自己モデル内部の関数 $Predict$ は，この環境状態の情報を教師あり学習することで得られる．

さらに関数 $Interpret$ を獲得する．得られた環境状態の隣り合う 3 フレームをまとめた組 (s_{t-2}, s_{t-1}, s_t) に対し主成分分析を用いて次元削減した上で，k-means 法によってクラスタリングを行う．この操作は，自身の行動による環境の変化を分類したものと考えられる．

次に，得られた各クラスタ c について，他者からのメッセージの各ビットの生起確率を計算し，この確率に基づいて自己モデルが生成するメッセージの各ビットを定める．クラスタを表現するメッセージの k 番目のビット $m(k)$ を，

$$m(k) = \begin{cases} 1 & (u > p(m'_t(k) = 1 | c \ni (s_{t-2}, s_{t-1}, s_t))) \\ 0 & otherwise \end{cases} \tag{1}$$

で定める．ここで u は $[0, 1)$ の一様乱数，$p(m'_t(k) = 1 | c \ni (s_{t-2}, s_{t-1}, s_t))$ は，環境状態の組がクラスタ c に分類される時，熟練エージェントからきたメッセージ m' における k 番目のビットが 1 である確率である．

この操作は，未熟エージェントが自分の振る舞いに対して，熟練エージェントのメッセージをベースにラベル付けを行い，自己解釈としているものと考えられる．

2.4 初期熟練エージェント

このシステムでは一番最初に，自己モデルを持つ熟練エージェントを用意する必要がある．今回は人の手により大雑把な戦略（加/減速する，左/右へ向かう，時計/反時計回りに姿勢を立て直す）を与え，これを初期熟練エージェントとする．このメッセージをもとに前述の操作を行うことで，人間の戦略を基礎とする自己モデルが構築されると考えられる．

3 実験概要

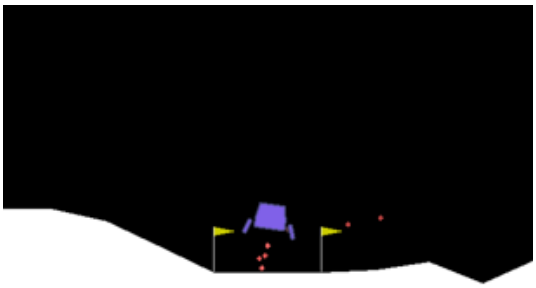

図 3: Lunar-Lander

提案する自己モデル構築法の検証のため，実験を行う．エージェントに操作させる環境として OpenAI の gym[4] を介し，Lunar-Lander (図 3) というゲームを利用する．これはロケットを地面に対し少ない衝撃で着陸させることを目的とするゲームで，エージェントは 4 つのコマンド (左/右/下へのジェット噴射，何もしない) でロケットの姿勢を制御することで，クリアすることを目指す．このゲームを選択した理由は，減速，旋回，移動など複数の戦略を考慮できるような，ある程度の複雑性を持っているからである．

今回の手法で意味のある自己モデルが構築されていれば，それによるメッセージを利用した場合の学習は，利用しない時の学習と比べて早く進むと考えられる．そこで，未熟エージェントの制御モデルの学習の進み具合をもって，構築した自己モデルを評価することにする．

4 おわりに

本稿では，熟練エージェントと未熟エージェントの二者を用いて，自己の制御モデルが出力する行動に対し解釈を与えるような自己モデルの構築法を提案した．

今回の実験ではゲーム環境内にエージェントが一体しかおらず，環境の変化が全て自身の行動の結果である，という仮定を前提としている．しかし環境が複数のエージェントで構成されるとき，自身の行動の結果と環境の変化は一対一の対応ではなくなる．そのため複数エージェント条件下では，環境の変化を自身の行

動によるものと他者の行動によるものに分類する機能が必要とされる．

また今回の提案では，自己モデルが逆に制御モデルに働きかけるような機構や，メッセージ自体を解釈する機構にはあまり触れていない．しかしメッセージの生成と解釈は表裏一体であり，今後は文脈などを踏まえて他者からのメッセージを解釈するモデルも考慮したい．

参考文献

[1] Whiten, A.: When does smart behaviour-reading become mind-reading?, in Carruthers, P. and Smith, P. K. eds., *Theories of theories of mind*, pp. 277292, Cambridge University Press (1996)

[2] 朴嵩哲: 理論説 vs. シミュレーション説 : 両説は結局どこが違うのか?, 哲学・科学史論叢, Vol. 13 pp. 123-167, 東京大学教養学部哲学・科学史部会 (2011)

[3] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostro- vski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis. Human-level control through deep reinforcement learning, *Nature*, Vol. 518, No. 7540, pp. 529533 (2015)

[4] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv*, preprint arXiv:1606.01540 (2016)