教師あり学習を用いた人狼知能の行動選択 ～全結合ニューラルネットワークを用いた予備実験～

高橋 篤剛[1,a)]　鶴岡 慶雅[2]

概要：人工知能研究の題材として、人狼ゲームが注目されている。機械学習により役職を推定する先行研究は数多いが、実際の行動選択を行う研究は少ない。従ってこの研究の目的は、機械学習を用いて人狼ゲームでうまい行動を選択できるエージェントを作成することである。全結合ニューラルネットワークを用いた教師あり学習で実験を行ったが、何も行動しないことが最善と認識されて有意な結果を得ることはできなかった。状態表現の改良などが今後の課題である。

Supervised Learning for Action Selection in the Game of Werewolf: a Preliminary Experiment with Fully-Connected Neural Networks

Atsutaka Takahashi[1,a)]　Yoshimasa Tsuruoka[2]

Abstract: There are many studies on the game of werewolf as the subject for artificial intelligence. Although many previous studies focused on estimating the positions, few addressed actual action choices. Therefore, the purpose of this study is to create an agent that can select actions to successfully play the game of werewolf using machine learning. This experiment was conducted using supervised learning with a fully coupled neural network, but it was recognized as a best choice to take no action, so the results were unsatisfactory. One if the important issues for future research is the improvement of state representation.

1. はじめに

技術の発展が著しい現代において、日々の暮らしをより便利にするために人工知能の研究が行われている。これまで人間が行っていた仕事を人工知能が肩代わりすることで人間の労働コストを削減したり、人間だけでは出来なかった目標を実現したり、などの効能があるため人工知能の研究は非常に意義がある。

人工知能の研究題材として、よくゲームが扱われる。ゲーム内の環境は実世界に比べて状態・行動表現が単純化されていて、また容易に何度もシミュレーションができるため、研究しやすいからである。AlphaGo [1] は完全情報ゲームの一つである囲碁において、人間を超える人工知能を実現した。この次にもう少し複雑な研究対象として、不完全情報ゲームである人狼ゲームが注目されている。

人狼ゲームは多人数不完全情報コミュニケーションゲームである。このゲームでは複数のエージェントが会話をすることで進行する。会話を通して自分以外のエージェントの考えを推察する必要があるために、不完全情報ゲームとして扱われる。不完全情報ゲームとはプレイ中に他プレイヤーの所有する情報を完全には把握できないゲームのことで、完全情報ゲームに比べて人工知能の作成が難しいことが多い。時には相手を騙すような発言をしたり、嘘を見破って相手の役職を推定したりする所に人狼ゲームの複雑さが存在し、これがこのゲームを研究する動機である。

人狼ゲームをプレイできるエージェントの開発をするために人狼知能プロジェクト [2] が立ち上がり、開発したエージェントでゲームをシミュレートするためのプラットフォーム [3] が提供されている。しばしば人狼知能大会

[1] 東京大学工学部電気電子工学科 Department of Electrical and Electronic Engineering, The University of Tokyo
[2] 東京大学工学部電子情報工学科, 東京大学大学院情報理工学系研究科電子情報学専攻 Department of Information and Communication Engineering, Graduate School of Information Science and Technology,The University of Tokyo
[a)] atsutaka@logos.t.u-tokyo.ac.jp

が開催され、大会に出場したエージェントのソースコードとログデータも公開されて、研究・開発がしやすくなっている。

本論文では人狼知能プロジェクトに基づいて人狼ゲームを研究する。具体的には、全結合ニューラルネットワークによる教師あり学習を用いて、このゲームにおける行動選択が可能なエージェントを作成する。

2. 背景

2.1 人狼ゲームのルール

本節では人狼ゲームの研究にあたって、このゲームのルールを確認する。最初に各エージェントに役職が与えられ、人間陣営と人狼陣営に分けられる。与えられた役職は他エージェントには秘密になっているが、エージェント同士が会話をすることで相手の役職を推定していく。会話の後で人狼と思わしきエージェントを投票で決めて処刑する。人狼は処刑されないよう嘘を混ぜてうまく会話をする必要があり、逆に人間は早く人狼を処刑するために見極める必要がある。処刑されなかった人狼は、その後で任意の人間を一人襲撃する。処刑・襲撃されたエージェントは死亡扱いになり退場する。会話→処刑投票→襲撃が 1 日の基本的な流れであり、人間が全員死亡するか人狼が死亡するとゲームが終了する。これを図示すると、図 1 のようになる。

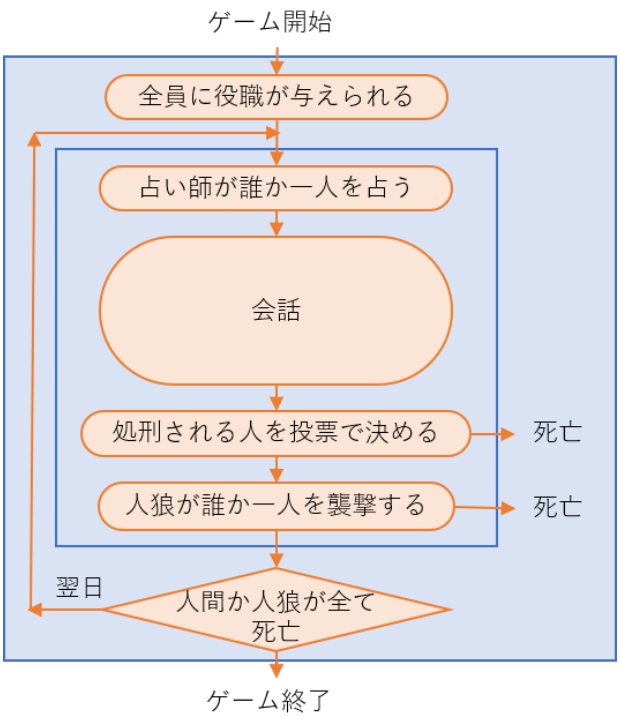

図 1 人狼ゲーム (5 人用) のプレイチャート

ゲームをプレイするエージェントの人数は 5 人用や 15 人用など何種類かあるが、今回の実験では学習を簡単にするために 5 人用を扱った。5 人用人狼ゲームにおける役職を表 1 に示す。役職名については、一般の人狼ゲームでよく使われる名前と人狼知能のプロトコルで定められている名前を記しておく。

表 1 役職の種類 (5 人用)

役職	説明
村人 (villager)	普通の人間。特殊能力を持たない。2 人いる。
人狼 (werewolf)	人狼。人間の敵。毎日 1 回、襲撃ができる。
占い師 (seer)	人間。毎日 1 回、占いができる。
狂人 (possessed)	人間だが、会話中は人狼の味方として行動する。

次にエージェントの行動を表 2 に示す。このうち襲撃は人狼だけの、占いは占い師だけの行動である。会話は本来何を喋っても構わないが、この実験では表 3 に示す通り人狼知能のプロトコルで定められた発言だけを行うようにする。会話中に決められた回数発言するか、全員が over() を発言した場合に 1 日の会話が終了する。

表 2 行動の種類 (5 人用)

行動	動作
会話 (talk)	他エージェントとコミュニケーションする。
投票 (vote)	処刑 (execute) されるエージェントを決める。
襲撃 (attack)	処刑投票の後に、指定したエージェントを死亡させる。
占い (divine)	会話の前に、指定したエージェントが人間か人狼か識別できる。

ここで {target} は発言対象のエージェントを表す。{role} は [村人, 人狼, 占い師, 狂人] の中で、{species} は [人間, 人狼] の中で該当するものを表す。{text} は任意の発言を表す。{day} は指定した日を表す。skip を連続で発言した場合、over を発現したものとして扱われる。

表 3 人狼知能のプロトコルで定められた会話中の発言

発言	動作
comingout	target の役職が role であると宣言する。
estimate	target の役職が role ではないかという推察を話す。
divination	target を占うことを宣言する。
divined	target を占った結果、species だったと報告する。
vote	自分は target に処刑投票をすると宣言する。
request	target に text をするよう要求する。
agree	day の target の text に賛成する。
disagree	day の target の text に反対する。
skip	何も話さずに、他のエージェントの会話を聞く。
over	その日の発言を終える。

2.2 先行研究

人狼知能大会で使用されるエージェントはルールベースで記述されたものが多い。現状では人間の手で人狼ゲームの振る舞い方をコンピュータに教え込んだ方が簡単に強いエージェントが作成できる。一方でいくつかの先行研究では、機械学習を用いて人狼ゲームのある一部分の仕事をこなしている。

梶原らの研究 [4] では、単層の教師あり学習である SVM [5] を用いて、観測された状態から人狼を推定する実験を行った。状態表現としてゲーム内の日にち、役職宣言した占い師の数、及び注目エージェントについて被占い結果、役職宣言、(占い師を宣言している場合) そのエージェントの占い報告、投票宣言回数を用いた。学習モデルの出力は、注目エージェントが人間か人狼かの 2 クラス分類とした。

源らの研究 [6] では、上記から発展させて LSTM [7] を含む多層モデルで教師あり学習を行い、人狼推定する実験を行った。LSTM は過去の情報を考慮するネットワークである。状態表現として襲撃されたエージェント、処刑されたエージェント、役職宣言したエージェントとその役職、会話中に判明した占い結果、各エージェントの投票宣言、投票要求、役職推察を用いた。出力は各エージェントについて人狼であるかの判別とした。

王らの研究 [8] では、投票・襲撃・占いの各行動において強化学習を用いて行動選択をする実験を行った。上記とほぼ同様の状態空間を取り、DQN [9] によって投票・襲撃・占い時にどのエージェントを選ぶかの行動選択を学習した。

2.3 教師あり学習

ルールベースよりも一般性を持った人工知能を実現するために機械学習を行う。この実験では機械学習の中でも教師あり学習を用いてエージェントを設計した。教師あり学習では、入力データとそれに対して正しい出力となる正解ラベルを与えて学習を行う。入力データを $x \in \mathbb{R}^n$、それに対するコンピュータの出力を $f(x) \in \mathbb{R}^m$、正解ラベルを $y \in \mathbb{R}^m$ として、誤差関数 $E(f(x), y)$ を計算する。コンピュータが持つモデル f は、一般に重みパラメータ $w \in \mathbb{R}^n * \mathbb{R}^m, b \in \mathbb{R}^m$ を用いて $f(x) = xw + b$ のような構造をとる。次に誤差関数の勾配) $\frac{\partial E}{\partial x}$ をコンピュータに還元して重み w, b を修正することで、コンピュータが持つモデル f を学習することができる。誤差関数は $f(x)$ と y が近いほど小さくなるものを採用し、例えば出力ラベルを分類する問題ではクロスエントロピー誤差が用いられる。

3. 予備実験

3.1 目的

2.2 節で例示したように、先行研究の多くは会話などで得られた情報を用いて他エージェントの役職推定をすることを研究の目的としている。確かに相手の役職を高精度で識別できれば人狼ゲームを有利に進めることができるため、それらの研究の価値は大きい。一方でこのゲームの終了条件は人間か人狼のいずれかが全て死亡した時であるので、会話中に自分が殺されないように振る舞うことは勝利に直接的に貢献する。それにも関わらず会話中の行動に着目した先行研究は少ない。従ってこの研究の目的は、会話中の行動を中心に、ゲームを最後まで生き残ることができるような行動選択が可能なエージェントを作成することとする。

3.2 内容

先述した目的に基づき、教師あり学習を用いて人狼知能エージェントを作成し、その性能を評価した。学習モデルは図 2 のように設計した。エージェントの実装は Python 用の人狼知能プラットフォーム [3] を、モデルの実装は tensorflow.keras [10] を使用した。学習には 2018 年 CEDEC 人狼知能大会から、5 人用である 15299 ゲームのログデータを利用した。まず各ゲームのログデータから、ゲームの最後まで生き残ったエージェントを調べた。そしてそのエージェントがゲームの中で取った行動とその時の状態を抽出してサンプルデータとした。エージェントが観測する状態表現を表 4 に、行動空間を表 5, 表 6 に示す。サンプルデータの前 8 割を学習データ、後ろ 2 割をテストデータとして教師あり学習を行った。この時エージェントの役職と行動の種類ごとにサンプルデータを区別して、異なるモデルとして学習した。

次に作成した学習モデルで実際に人狼ゲームをシミュレートして性能を検証した。このモデルを搭載したエージェント 1 人と人狼知能プロジェクトが提供するサンプルエージェント 4 人で人狼ゲームを行った。エージェントの役職ごとにゲームは 1000 回ずつ行い、その結果から実験エージェントがゲームの最後まで生き残った確率と、処刑投票の時に人狼に投票した確率を集計した。また対象実験として、会話は常に skip() を選択し、それ以外の行動ではランダムに対象を選択するランダムエージェントを用いて同様のシミュレーションを行った。

より柔軟な行動選択を実現するために、学習モデルは 3 層ネットワークで表現した。この実験は分類問題であるため、誤差関数としてクロスエントロピー誤差を使用した。また過学習を防ぐため、誤差関数に L2 正則化項を加えた。

ここで {agent} は行動を起こすエージェントを、{target} は行動対象のエージェントを表す。{role} は [村人, 人狼, 占い師, 狂人] の中で、{species} は [人間, 人狼] の中で該当するものを表す。{day}、{turn} はそれぞれ int 型で与えられる。

状態表現は先行研究 [6] を参考に、全部で 267 次元の特徴量で記述した。day と talk_turn 以外はすべて one-hot ベクトル、即ち 0 or 1 で表した。シミュレーション時には、

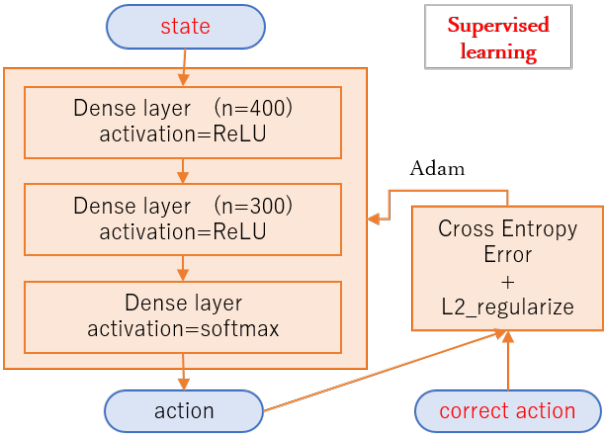

図 2 教師あり学習モデル

表 4 状態表現

次元	特徴量	説明
0	day	日にちが {day} である。
1-25	vote	{agent} が {target} に処刑投票した。
26-30	execute	{target} が処刑された。
31-35	attack	{target} が襲撃された。
36-45	divine	(占い師のみ){target} を占った結果 {species} と判明した。
46-65	talk_comingout	{agent} が自分の役職が {role} であると宣言した。
66-115	talk_divined	{agent} が {target} を占った結果 {species} だったと報告した。
116-140	talk_vote	{agent} が {target} に投票を宣言した。
141-165	talk_request_vote	{agent} が {target} への投票を要求した。
166-265	talk_estimate	{agent} が、{target} の役職が {role} だという推察を話した。
266	talk_turn	その日の会話で {turn} 回の発言がされた。

表 5 会話における行動空間

次元	特徴量	説明
0	talk_skip	何もしない。
1-4	talk_comingout	自分の役職が {role} であると宣言する。
5-14	talk_divined	{target} を占った結果 {species} だったと報告する。
15-19	talk_vote	自分は {target} に処刑投票すると宣言する。
20-24	talk_request_vote	周囲に {target} に処刑投票するよう要求する。
25-44	talk_estimate	{target} の役職は {role} だという推察を話す。

表 6 投票・襲撃・占いにおける行動空間

次元	特徴量	説明
0-4	choice	{target} を選択する。

エージェントに情報が与えられる毎に該当する one-hot ベクトルを更新することとした。

会話における行動空間は表 3 の発言項目の中から、ゲームとして有効と思われる発言に絞り 45 次元で表現した。ログデータを読むときに、この表にない行動は全て skip として扱った。シミュレーション時には最大値をとる行動を選択することとした。

会話以外の行動では、5 人のエージェントから 1 人を選択することとした。

3.3 結果

役職・行動の種類ごとにモデルを学習した。最終的な精度を表 7 に示し、64 個のデータをミニバッジとして 10000 回ランダムサンプリングして学習したときのロスの推移を図 3 に示した。どの役職とも会話モデルの学習精度が比較的低かった。ただし会話の行動空間は 45 通りあるので、少なくとも無作為に行動選択しているわけではないと言える。

表 7 モデル別、教師あり学習の精度

役職	行動	訓練データ	テストデータ
villager	talk	68.4%	66.2%
	vote	83.0%	82.9%
werewolf	talk	74.3%	75.5%
	vote	77.7%	73.9%
	attack	71.9%	65.5%
seer	talk	64.3%	62.3%
	vote	84.7%	84.7%
	divine	90.4%	90.9%
possessed	talk	65.1%	64.3%
	vote	83.9%	82.5%

学習したモデルを用いてシミュレーションをした結果を表 8 に示す。alive rate はゲームを最後まで生き残った確率、vote は各処刑投票で人狼に投票できた確率を表す。どの役職でも実験エージェントとランダムエージェントとの違いがみられなかった。そこでシミュレーションのログデータを見たところ、会話時に skip しか選択できていなかった。

4. おわりに

この実験では全く効果的な結果が得られなかった。教師あり学習自体はできたものの、会話中に skip ばかり選択していた。これはログデータを読むときに例外的な行動を全て skip で置き換えるように実装したことに所以すると考えられる。従って状態空間をより詳細に捉える必要があ

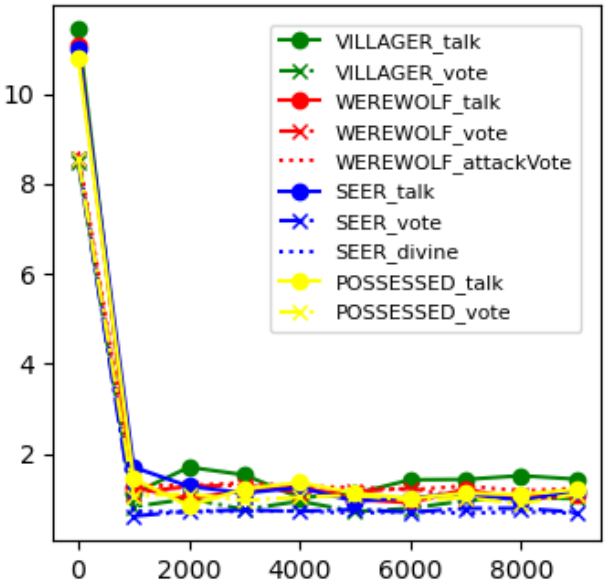

図 3 教師あり学習のロスの推移

表 8 シミュレーション結果

役職	指標	実験	ランダム
villager	alive rate	45.6%	42.8%
	vote(day1)	24.6%	24.2%
	vote(day2)	50.6%	49.8%
werewolf	alive rate	31.4%	34.4%
	vote(day1)	0%	0%
	vote(day2)	0%	0%
seer	alive rate	36.1%	33.1%
	vote(day1)	24.4%	24.1%
	vote(day2)	48.9%	48.5%
possessed	alive rate	50.6%	49.1%
	vote(day1)	24.1%	24.3%
	vote(day2)	50.4%	50.9%

る。ただし状態空間を広げすぎると学習をしづらくなるので、重要な情報だけ別に入力するなどのモデル改良も並行する必要がある。

会話中に何も話さないのが安定択として認識されたという解釈もできる。確かに目立たなければ処刑や襲撃の対象に選ばれにくいので、生存確率は上がる。しかしこれは人狼ゲームをプレイ出来ているとは言えない。生き残ったエージェントを選んで教師あり学習をするよりも、合理的な行動に報酬を与えて強化学習をする方が目的に沿える可能性がある。

人狼ゲームを正しくプレイできるエージェントが勝てるエージェントとも限らない可能性がある。つまり評価自体を生存確率ではなく合理的な行動を取れた確率、客観的な信用度などにする方がいいとも考えられる。

5 人用の人狼ゲームでは状態が分かってくる前に終了してしまうため、学習が難しくなっていることも原因として考えられる。従って 15 人用の人狼ゲームでも実験する必要がある。

参考文献

[1] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneersheivam, Marc Lanctot, Sander Eieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the Game of Go with Deep Networks and Tree Search. Nature 529, 484-489(2016)

[2] 人狼知能プロジェクト. http://aiwolf.org/

[3] 人狼ゲームプラットフォーム (Python 用). https://github.com/ k-harada/AIWolfPy

[4] 梶原健吾, 鳥海不二夫, 稲葉通将, 大澤博隆, 片上大輔, 篠田孝祐, 松原仁, 狩野芳伸: 人狼知能大会における統計分析と SVM を用いた人狼推定を行うエージェントの設計. 人工知能学会全国大会論文集,Vol. JSAI2016, pp. 2F412F41 (2016)

[5] V.N.Vapnik, A.Ya.Lerner. Pattern Recognition Using Generalized Portraits. Avtamarika I Telerncklanjka, VoI.21,No.6, pp.774-780(1963).

[6] 源智也, 松原仁: Long Short Term Memory による複数人の人狼推定. ゲームプログラミングワークショップ 2019, pp.126-129 (2019).

[7] Sepp Hochreiter, J urgen Schmidhuber, Fakult at fur Informatik. Long Short Term Memory. Neural Computation 9(8):1735-1780 (1997).

[8] 王天鶴, 金子知適: 人狼ゲームエージェントにおける深層 Q ネットワークの応用. ゲームプログラミングワークショップ 2018, pp.16-22(2018).

[9] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, Martin Riedmiller. Playing Atari with Deep Reinforcement Learning. NIPS Deep Learning Workshop 2013.

[10] tensorflow. https://www.tensorflow.org/?hl=ja