

公立はこだて未来大学 2020 年度 システム情報科学実習
グループ報告書

Future University-Hakodate 2020 System Information Science Practice
Group Report

プロジェクト名

AI するディープラーニング

Project Name

AI Love Deep Learning

グループ名

グループ A

Group Name

Group A

プロジェクト番号/Project No.

12-A

プロジェクトリーダー/Project Leader

松田 顕 Ken Matsuda

グループリーダー/Group Leader

小田中 嵐 Arashi Odanaka

グループメンバ/Group Member

松田 顕 Ken Matsuda

小田中 嵐 Arashi Odanaka

西村 和馬 Kazuma Nishimura

工藤 光矢 Koya Kudo

根上 樹 Tatsuki Neagari

木村 高之 Takayuki Kimura

指導教員

竹之内 高志 寺沢 憲吾 香取 勇一 佐々木 博昭 片桐 恭弘

Advisor

Takashi Takenouchi Kengo Terasawa Yuichi Katori Hiroaki Sasaki Yasuhiro Katagiri

提出日

2021 年 1 月 14 日

Date of Submission

January 14, 2021

概要

近年、ディープラーニングの技術を用いた開発が盛んに行われている。従来の機械学習よりも複雑な問題に対して対応できるこの技術は、様々な分野で応用されている。その中で、将棋や囲碁などプレイヤーがすべての情報を観測できる完全情報ゲームをプレイする人工知能が開発されている。将棋、囲碁においてはいずれも人工知能がトッププロの人間に勝利し、人工知能が人間を超越したと言える。しかし、麻雀やポーカーなどのように情報のすべてを把握することができない不完全情報ゲームにおいては、人工知能は未だ人間を超越したとは言えず、研究が進められている。

本グループでは不完全情報ゲームのひとつである「汝は人狼なりや？」通称人狼ゲームにおいて、人狼ゲームをプレイするエージェントの従来とは異なる新しい手法を考案し、人狼知能プロジェクトの主催する人狼知能大会で好成績を残すエージェントの開発を行う。従来のモデルは、役職推定を機械学習、行動選択をルールベースで行うことが主流である。本グループでは、これとは異なる特色のあるエージェントの開発を目指す。

前期では開発環境の構築、必要な知識の共有、作成するエージェントの方針決定・使用する機械学習モデルの検討を行った。後期の活動ではエージェントの開発を進め、勝率を高めていく必要がある。

後期では、機械学習のモデルにニューラルネットワークを採用し、ログデータから得られる様々な情報を駆使して人狼の組み合わせを予想する人狼推定モデルを構築した。さらに、行動決定モデルには強化学習の一種であるDQN(Deep Q-Network)を採用し、状況に応じて最適な行動を選択するエージェントを開発した。

まず、過去の人狼知能大会のログデータを解析し、人狼推定に使用するデータの抽出作業を行った。得られた304次元のデータを特徴量とし、Pythonの機械学習ライブラリであるKerasを使用して、ニューラルネットワークを用いた人狼推定モデルを構築した。その結果、正しい人狼の組み合わせを約63%で推定できるモデルの構築に成功した。

次に、人狼推定モデルに使用した304次元のデータを用いて、DQNで行動推定モデルを構築した。具体的には、人狼ゲームにおいて自陣に有利に働くような行動に正の報酬、不利に働くような行動に負の報酬を与え、報酬が最大となるような行動を選択するように学習させた。

これら2つのモデルを用いて人狼推定と行動決定を行いながら戦うエージェントの開発に成功した。勝率は43.3%であり、さらなる改良が見込まれる。

キーワード ディープラーニング, 人狼知能, ニューラルネットワーク

(※文責: 小田中嵐)

Abstract

In recent years, active development using Deep Learning technology has been processing. This technology can solve more complex problems than conventional machine learning, applied to a wide range of fields. In recent years, In this field, artificial intelligence has been developed to play called “Perfect Information Game”, such as Shogi and Go, players can observe all information in these games. As for Go and Shogi, it can be said that artificial intelligence has surpassed humans, as artificial intelligence has defeated their respective professionals. However, the same cannot be said for games that are “Imperfect Information Game” like mahjong or poker, and research is being done every day.

This group will develop “Agent” (Refers to an artificial intelligence player in a werewolf game) with different methods than conventional agents in one of the “Imperfect Information Game”, the Werewolf Game. In addition, the goal is to bring the agent’s capabilities to bear in competitions organized by “Artificial Intelligence based Werewolf”, a research organization. conventional models are predominantly machine learning for position estimation and rule-based for action selection. On the other hand, our group plans to develop an unique agent, unlike that.

In the first semester, we built a development environment, shared knowledge, decided on a policy for agents to be created and examined the machine learning model to be used. As a policy, we planned to adopt neural networks as a model for machine learning and use this technology for everything from role position estimation to action selection.

In the second semester, we adopted neural networks for the machine learning model and constructed the “Werewolf Estimation Model” that predicts the combination of werewolves using various information obtained from log data. On the other hand, we adopted Deep Q-Network (DQN), a type of reinforcement learning, for the “Action Decision Model”, and developed an agent that selects the best action depending on the situation.

Firstly, we analyzed the log data of past competitions and extracted the data to be used for werewolves estimation. Using the 304-dimensional data obtained from log data as features, we constructed the “Werewolf Estimation Model” using neural networks by Keras, a machine learning library in Python environment. As a result, we succeeded in constructing a model that can estimate the correct combination of werewolves with about 63%.

Next, using the same features and the results of the “Werewolf Estimation Model”, we built the “Action Decision Model” by DQN. Specifically, we set positive rewards to actions that would work in favor of our player’s team and negative rewards to actions that would work against one in a werewolf game, and trained the model to select the action that maximized the rewards.

Using these two models, we have developed the agent “FUN Wolf” that can estimate werewolves and make action decisions. The winning rate was 43.3%, and further improvement of accuracy is expected.

Keyword Deep Learning, Werewolf Intelligence, Neural Networks

(※文責: 根上樹)

目次

第 1 章	はじめに	1
1.1	プロジェクト概要	1
1.2	グループ概要	1
1.3	背景	1
1.4	目的	2
1.5	期待される効果	2
第 2 章	課題設定	3
2.1	従来の主流な方法	3
2.2	具体的な課題設定（関連する講義/利用すべき習得技術・既存技術）	3
2.3	短期目標と長期目標	4
第 3 章	活動内容	5
3.1	前期	5
3.1.1	活動の詳細	5
3.1.2	活動の問題点	6
3.1.3	解決方法	6
3.1.4	後期への引継ぎ	6
3.2	後期	6
3.2.1	活動の詳細	7
3.2.2	活動の問題点	9
3.3	共同開発環境	10
3.3.1	Git	10
3.3.2	GitHub	10
3.3.3	Anaconda3	11
第 4 章	人狼知能の詳細	12
4.1	人狼知能プロジェクトについて	12
4.2	人狼知能大会について	12
4.3	プロトコル部門について	12
4.4	人狼ゲームの詳細	13
4.4.1	人狼ゲームの概要	13
4.4.2	役職	13
4.4.3	ゲームの流れ	14
第 5 章	エージェントの概要	16
5.1	概要	16
5.2	ニューラルネットワークについて	17
5.3	エージェントの詳細	18

5.4	機械学習モデルの詳細	19
5.4.1	開発環境	19
5.4.2	人狼推定モデル	19
5.4.3	行動決定モデル	22
5.5	評価実験	24
5.6	考察	24
第 6 章	活動結果	26
6.1	詳細な活動結果	26
6.1.1	前期	26
6.1.2	後期	26
6.2	中間発表	27
6.2.1	中間発表の準備	27
6.2.2	評価結果	27
6.2.3	評価の考察	28
6.3	最終成果発表	29
6.3.1	最終成果発表の準備	29
6.3.2	評価結果	29
6.3.3	評価の考察	30
第 7 章	個人活動	32
7.1	松田顕	32
7.1.1	前期	32
7.1.2	後期	32
7.2	小田中嵐	33
7.2.1	前期	33
7.2.2	後期	33
7.3	西村和馬	34
7.3.1	前期	34
7.3.2	後期	34
7.4	工藤光矢	35
7.4.1	前期	35
7.4.2	後期	35
7.5	根上樹	36
7.5.1	前期	36
7.5.2	後期	36
7.6	木村高之	37
7.6.1	前期	37
7.6.2	後期	37
第 8 章	まとめ	38
8.1	プロジェクト活動の成果	38
8.2	改善点	39

第 1 章 はじめに

1.1 プロジェクト概要

近年、ディープラーニングの技術を用いた開発が盛んに行われている。この技術は、人間が行う仕事をコンピューターに学習させる機械学習の手法の一つであり、従来の機械学習と比べてより複雑な問題に対して適用することが可能となった。本プロジェクトでは、この最新技術を活用し、新たなアプローチによる人狼知能の開発及び、自然言語処理を用いた競馬分析、画像解析を用いた感染症対策支援の 3 つの問題を解決する方法を模索する。

(※文責: 西村和馬)

1.2 グループ概要

本グループ「人狼知能」では、ディープラーニング関連技術を用いて、人狼ゲームをプレイできるエージェントの開発を、従来とは異なるアプローチにより行う。人狼知能プロジェクト [1] の主催する大会に出場し、好成績を収めることが、本グループの目標である。

(※文責: 西村和馬)

1.3 背景

2015 年に情報処理学会がコンピューター将棋プロジェクトにおいて、トッププロ棋士に勝つという目的を事実上達成したとして、コンピューター将棋プロジェクトの終了宣言を行った [2]。また、2016 年には GoogleDeepMind 社が開発した AI である AlphaGo が、囲碁の世界トップクラスの棋士をやぶり、完全情報ゲームで最も難しいものと言われている囲碁において、人工知能が勝利を収めた。したがって、将棋や囲碁など、プレイヤーがゲームの情報をすべて観測できる完全情報ゲームにおいて、人工知能が人間を超越したといえる。その一方で、麻雀やトランプのポーカー、大富豪など、情報を得るために推論を必要とする不完全情報ゲームでは人間を超越したとは言えず、不完全情報ゲームについての AI を研究している多くの研究分野が存在する。その一例として、不完全情報コミュニケーションゲームである「汝は人狼なりや？」通称、人狼ゲームの研究を行っている人狼知能プロジェクト [1] があり、人狼知能セミナーや人狼知能大会などを開催している。また、人工知能を支える技術として機械学習があるが、機械学習で用いられるプログラミング言語として近年、Python 言語が人気を博している。その理由として、Python 言語はコードが簡単でわかりやすく、身近なサービスにも使われており、信頼性が高いといった特徴がある。また、Python 言語には機械学習に関するライブラリやフレームワークが多数存在し、数値計算、科学計算などを行う NumPy, SciPy, 多くの機械学習アルゴリズムが実装されている scikit-learn などがある。このように Python 言語を用いて機械学習を行うことが出来るが、Python 言語だけでなく、Java 言語や C++ 言語など、他の言語でもライブラリなどを用いて機械学習を行うことが可能である。したがって、近年は機械学習を行うための環境が非常に整っており、学生でも情報科学、機

機械学習の知識さえあれば、人工知能の研究に取り組むことが可能である。こうした背景から、我々はグループで話し合った結果、「人狼をプレイできるエージェント」を構築することを目的とし、そのために機械学習が可能なライブラリなどを用いて、人狼知能エージェントの開発を行っていく。

(※文責: 工藤光矢)

1.4 目的

我々人狼知能グループは、機械学習を用いて「人狼をプレイできるエージェント」を既存のエージェントとは違ったアプローチから構築することを目的としている。また、そのエージェントで、人狼知能プロジェクト [1] の主催する、人狼知能大会に出場し、エージェントの運用を行う。既存のエージェントは役職推定と行動決定の2つのステップから成り、それぞれ役職推定は SVM などの機械学習、行動決定はルールベースで行われているエージェントが多い。後期で考えられた既存のエージェントとは違ったアプローチの具体案としては、役職推定をニューラルネットワークで行い、行動選択を Deep Q-Network (DQN) で行うことを考えている。

(※文責: 工藤光矢)

1.5 期待される効果

我々が本グループ活動を遂行することによって、さまざまな効果が期待される。まず、将棋や囲碁などの完全情報ゲームにおいて、人工知能が人間を超越したといえるが、その一方で、麻雀やトランプのポーカー、大富豪などの不完全情報ゲームでは人間を超越したとは言えず、研究分野として未知の世界が広がっている。したがって、我々が人狼知能プロジェクト [1] の主催する人狼知能大会に出場することによって、不完全情報ゲーム研究の振興を図ることが期待できる。また、「人狼をプレイできるエージェント」を既存のエージェントとは違ったアプローチから構築するため、人狼知能研究に貢献ができると考えられる。

そのほかの効果として、将棋界において、将棋 AI が新たな手や戦法を発見したという例がいくらか見受けられる。また、将棋 AI は現在、人間と共存する段階に入っており、実際、自身の将棋の研究に将棋 AI を用いているプロ棋士がいくらか見受けられる。このように将棋 AI は将棋というゲーム自体に様々な影響を与えている。このことから、人狼知能においても新たな戦法などを発見する可能性を秘めており、人狼ゲーム自体に大きな影響を与えることが期待できる。

また、我々が本グループ活動を遂行することによって、グループ開発、機械学習、プログラミングなどの技術を有した、社会で活躍できる技術者として成長できるだけでなく、最終的には、成果物を最終発表会などで公表し、「人工知能はどのようなことが出来るのか」という知見を人々に与えることで、人工知能、機械学習技術の社会への導入を促進できることが考えられる。

(※文責: 工藤光矢)

第 2 章 課題設定

2.1 従来の主流な方法

人狼知能プロジェクトは過去に複数回の大会を開いており、前例としてそれらに参加したエージェントがある。7月に参加した人狼知能セミナー 2020 や、参考図書『人狼知能で学ぶ AI プログラミング』[3]、人狼知能に関する論文を元に従来の主流な手法について考察した。従来の人狼知能エージェントはその構成を役職推定と行動決定の 2 つに分けることができる。役職推定は、ゲーム中に得られるデータを元に他プレイヤーの役職を推定する部分である。この部分で機械学習を使用するのが主流である。参考図書では、基本的な人狼知能エージェントの作成で機械学習に SVM(サポートベクトルマシン)を使用している。行動決定は「投票」、「発言」などの人狼ゲームで行う行動を決める。行動選択のほとんどにルールベースが使用されており、役職推定の結果から投票先などを決定している。ルールベースとは、例えばエージェント A の役職推定の結果が人狼であった場合、エージェント A に投票するといったように設計することである。つまり、具体的な戦略は作成者によって決定されていることがわかる。

人狼知能プロジェクト [1] 主催の大会に参加しているエージェントは、この役職推定と行動決定に工夫を加えたものが多い。また、大会に参加するエージェントはその 1 つ前の大会のエージェントに強くなるような設計となっている傾向がある [4]。よって現在は機械学習による役職推定の精度を向上させること、役職推定の結果から最適な勝ち筋を作成者が考えることが人狼知能エージェントにおける主流であるといえる。

(※文責: 松田顕)

2.2 具体的な課題設定 (関連する講義/利用すべき習得技術・既存技術)

本グループの目標は人狼知能プロジェクト [1] 主催の大会において高成績を残し、かつ特色のある人狼知能エージェントを作成することである。その実現方法として、機械学習のモデルにはニューラルネットワークが最適であると判断し、役職推定から行動決定までを行うこととした。モデルの使用法として、入出力をすべて 1 つのモデルを使用する、自身の役職ごとにモデルを作成するなどいくつかの方法が検討されている。

目標を達成するための課題として、まず全員での知識共有がある。人狼知能プロジェクトそのものや、機械学習技術などについての学びを共有する。もう一つの課題として使用する言語、ライブラリなど共同開発環境を構築することがある。開発環境を整えておくことでより実践的かつ効率的な開発と学習を行っていく。

(※文責: 松田顕)

2.3 短期目標と長期目標

短期目標として、前期には前節の具体的な課題設定であるように知識の共有と共同開発環境の構築を並行して行っていく。また、参考図書『人狼知能で学ぶ AI プログラミング』[3] の輪読を進める。最終目標についても戦略などのアイデアを考えていく。より具体的には、まず輪読と調査を通して人狼知能プロジェクトについて知り (調査結果は 4 章)、人狼知能エージェントの作成方法について学ぶ。共同開発環境の構築は、使用する言語や Git などのツールを定め、全員がそれらを使いこなせるようにする。最終目標のためのアイデアとして前節で記述したような実現方法を考える。

前期時点での長期目標としては、人狼知能セミナーへの参加、人狼知能エージェントの作成、人狼知能大会への参加がある。人狼知能プロジェクト [1] 公式の勉強会である人狼知能セミナーが不定期で開催されている。逐次それらに参加をして情報と知識を集める。本グループの最終目標の達成には年に 2 回開催される人狼知能大会に参加することが必要条件となる。人狼知能大会に参加するために、人狼知能エージェントの作成を進める。

後期においては、ログデータの解析、人狼推定モデルの作成、行動決定モデルの作成、サーバー接続部の設計など、エージェント内における各要素の実装が短期目標として挙げられる。複数の班に分割して作業を行うことで効率の良い目標達成を狙う。

後期時点での長期目標について、年内に参加が可能である人狼知能大会が存在しないことを確認したため、来期以降の大会にエージェントを参加させるという方針へと定め直した。また引き続きエージェントの完成が最終目標として挙げられる。具体的な完成期限については、最終成果発表日までにエージェントの完成および評価実験を行う予定とした。

(※文責: 松田顕)

第 3 章 活動内容

3.1 前期

主に前期活動では、前提となる知識共有、前例の調査、手段の検討・選定を行った。6、7月にかけて輪読を行い、機械学習の基礎知識や、人狼知能に搭載すべきアルゴリズムの要件を確認した。7月以降は、人狼知能セミナー 2020 を通して大会成績・勝率を意識した、過去大会エージェントの調査・対策について議論を行った。夏季休暇前後には、中間発表での反応や個人での調査を踏まえて、エージェントの方向性の検討および実際に利用する機械学習モデルの選定を行った。例年と比較して活動期間が圧縮されたが、事前計画を緻密に立てたことでスムーズな活動が可能であった。一方全編通してオンライン上での活動であり、後述する問題点を抱える結果となった。

(※文責: 根上樹)

3.1.1 活動の詳細

知識共有のために輪読した参考図書は『人狼知能で学ぶ AI プログラミング』である [3]。主に人狼知能エージェントのフレームワーク解説が記述されており、人狼知能エージェントの構成の仕方について学習した。また機械学習全般の基礎知識にも言及されており、関連用語のグループ内での統一に用いた。

環境構築について、人狼知能エージェントが活動するためのプラットフォームを形成するための Java 言語、人狼知能エージェントそのものを構築するための Python 言語、作成物の共有や履歴管理のためリポジトリ管理ツールである Git および GitHub をそれぞれ利用するためのソフトウェアおよびライブラリを導入した。

また活動の一環として、2020 年 7 月 5 日に行われた人狼知能プロジェクト [1] 主催の人狼知能セミナー 2020 に参加した。現状の研究最前線では、役職推定に機械学習を搭載した、行動決定をルールベースで行うエージェントが主流であることを知ることができた。

前期活動の結果、作成するエージェントのおおまかな前提条件が定まった。まず、エージェント作成に使用する言語として Python 言語を用いることとなった。当初 (人狼知能セミナー 2020 参加以前) は Java 言語の利用を検討していたが、機械学習ライブラリの豊富さを加味した結果現在のものとなった。次にエージェントの行動決定にも機械学習の判断を導入することを決めた。大会成績を考えると、例にならったような行動選択にルールベースを用いたエージェントでは、有意な違いを算出するために、資源や知識が周囲と比較して不足していると判断した。機械学習モデルには、ニューラルネットワークを採用することに意見が一致した。ただし、入出力の範囲を定めるためにプロトタイプを作成したうえでの調整が必要であり、具体的な入力値・出力値の算出に関しては後期の作業予定となった。

(※文責: 根上樹)

3.1.2 活動の問題点

オンラインでのプロジェクト活動を通じた問題点が複数存在した。特に多人数での議論の際に、回線状況や音響設備などの影響で進行が停止する場合があった。また、一度に一つの会話しか行うことができない、同時に発声することを避けるために議論速度が低下する、といった問題も散見された。

活動内容についての問題点としては、スケジュールの過密が懸念される。9月現在までに基礎作業はほぼ完了している一方、エージェント作成の実作業に関してはほとんど進行していない。後期発表までの期間はそう長いものではなく、いずれかの期間に重大作業が集中することが考えられる。

(※文責: 根上樹)

3.1.3 解決方法

後期には活動3回につき1回、オフラインでの活動が認められる。重要な議論および注意共有はその機会に行うようにすることにより、問題点の解決を図る。また後期には技術的な製作活動が増加することを考えると、以前より記録に残りやすい文面を通じた共有を重視する必要がある。

更なるスケジュールの工夫・改善も必要であると考えられる。特にスケジュールの間延びを防ぐために、疑問点の共有や前提のすり合わせを積極的に行うことが望まれる。また徹底した役割分担を行い、特定の誰かの負担が大きくなりあまり計画がとん挫する、といった事態を防ぐことも重要である。

(※文責: 根上樹)

3.1.4 後期への引継ぎ

特にプロトタイプの作成が急を要する。ニューラルネットワークを利用した複数の形式のエージェントを比較することを予定している。当初の形式に囚われず、臨機応変なモデルを選択すべきであるという結論に至った。また残時間次第ではあるが、作成したエージェントの動作を視覚的に周知させることのできる仕組み・図解の作成を行うことも望まれる。

(※文責: 根上樹)

3.2 後期

後期は本格的に人狼知能エージェントの開発に取り掛かった。開発にはモデル班と設計班の2つに分かれて取り組んだ。一つのまとまったグループで作業を行うには、あまりにも実装しなければならない機能が多いと判断したためである。モデル班はログデータから人狼を推定する人狼推定モデルと状況や人狼推定モデルからの情報から次にとる行動を決める行動決定モデルの作成を担当した。設計班はモデル班が作成したモデルに適切な特徴量を渡し、出力に合わせた行動などの処理を行うようなエージェントの構築を担当した。モデル班には小田中・根上・工藤・西村が所属し、設

計班には松田・木村が所属した。

活動終盤にはモデル班・設計班を合流し、それぞれの作成物を組み合わせる作業を行った。また各種モデルの不具合修正や評価実験、考察も合流後に行う形となった。

活動に際して特殊な事情がない限り、水曜日を対面活動、金曜日をオンライン活動と定めた。前期での反省を踏まえ、複雑な作業や話し合い、評価実験をできる限り対面活動日に行うようにした。オンライン活動時には会議ツールに追加された新たな機能によって複数の部屋を立ち上げ、班ごとに分かれて作業することで議論・作業の速度をある程度十分に保つことができた。

なお夏季休暇中および後期初期には中間グループ報告書の執筆を行った。報告書の執筆を通して、現状の活動計画を見つめ直すことができた。特に複数の方向性のエージェントを作成するという予定に関して、あまり現実的ではないという方向性で話がまとまった。その結果、一つのエージェントに複数のモデルを搭載するという発想が生まれ、後期の予定をやや変更することとなった。

また成果発表後から1月上旬にかけて最終グループ報告書の執筆を行った。成果発表前に最終成果物の評価実験までを行うことができていたため、中間時よりスムーズな作業を行うことができた。グループ報告書の執筆の中で、成果発表で得られた感想・意見を共有した上での反省を行い、明確な反省点・改善点を表面化させることができた。

(※文責: 松田顕)

3.2.1 活動の詳細

モデル班

機械学習モデルを構築するにあたり、ディープラーニングのライブラリの1つである Keras を導入した。Keras は、ニューラルネットワークモデルの構築を比較的容易にするライブラリである。また数値データを高速に処理するため、計算支援ライブラリである NumPy、人狼知能サーバーとの連携を行うため、データ解析を支援するライブラリである pandas をそれぞれ導入した。導入した各種ライブラリについて理解を深めたのち、実作業工程へ突入した。

まず人狼推定モデルと行動決定モデルを構築するうえで、過去の人狼知能大会のログを解析し、教師用データ、学習用データとして使用した。過去大会のログは人狼知能プロジェクト公式サイト [1] でテキストファイル形式で公開されており、そこから後述のモデルを構築するために必要な 304 次元の特徴量を抽出した。今回は公開されているログの中で最も新しい「第 4 回人狼知能大会」の決勝戦のログデータを用いた。

ログデータ解析の終了後、ニューラルネットワークモデルを用いて「人狼推定モデル」を構築した。構築するモデルは、ログデータを解析した 304 次元のデータから人狼を推定する必要があるため、入力を 304 次元とし、出力を人狼の組み合わせの総和とするニューラルネットワークモデルとした（更なる詳細は「5.4.2 節 人狼推定モデル」を参照）。

人狼推定モデルの構築完了後、実際のゲーム中における行動を決定するための「行動決定モデル」を構築する段階に入った。形式として、Deep Q-Network (DQN) モデルによる強化学習を採用した。モデルの動作確認をする際にゲームサーバ環境が必要になったこともあり、より設計班との連携作業を強めることとなった（更なる詳細は「5.4.3 節 行動決定モデル」を参照）。

モデルの構築作業に期限を設け、以降は学習・運用実験を行った。各種モデルやシステムが最低限エラーを発生しないことを確認したのち、あらかじめ定めた実験方針に則って学習・運用を行った。

設計班

設計班の活動として、最初に Keras と pandas の学習を行った。Keras は機械学習を行う際に用いられる Python のライブラリであり、その軽量を理由に採用した。Keras を勉強することでモデル班が作成した機械学習モデルをエージェントにインポートし、そのモデルに必要な特徴量を渡し、出力を正しく受け取る流れを正しく実装できるようになった。Keras の学習は公式ドキュメント [5] を主な教材として用いた。pandas とはデータ解析に用いられる Python ライブラリであり、機械学習モデルに渡す特徴量を作成するために用いた。pandas の学習には Chainer のチュートリアル [6] を用いて行った。Keras と pandas の学習は、Colaboratory を用いて実践を通して行った。Colaboratory とは Google が提供するサービスの一つであり、ウェブブラウザ上で Python を実行することができる。環境構築が必要ないことや、共同編集が可能であることからライブラリの学習以外でも開発の際に実装前の動作確認でこのサービスを利用した。

開発には GitHub などを用いた。モデル班の機械学習モデルの作成と並行して開発を行い、モデル班の要望に合わせてエージェントに編集を加えていった。

人狼知能エージェントの作成には、人狼知能プロジェクトが提供している Python で作成された基本エージェントに機能を追加していく。正確な挙動を実現するために、基本エージェントで使用されているメソッドや変数について調査を公式ドキュメント [7] を参考に行った。以下がその結果である。

メソッド

initialize ゲームの初めに 1 度呼び出され、ゲームの初期化を行う。base_info, diff_data, game_setting が引数として与えられる。

update 1 日で複数回呼び出される。base_info, diff_data が引数として与えられる。

dayStart 1 日の初めに 1 度呼び出される。

変数

base_info 自分の役職、相手の役職、相手の状態、残りの発言、ささやき可能回数が格納された辞書型

diff_data agent, day, idx, text, turn, type をインデックスに持つ Pandas の DataFrame 形式。少し特殊であり、type の値によって agent, idx, turn の指す内容が変わる。

game_setting プレイヤー人数などゲームの基本情報が格納された辞書型

diff_data について具体的に解説を行う。type が “initialize” または “finish” の場合はゲームの開始または終了を示し、idx, agent には自分のエージェント ID が、text には finish の場合のみ自身の役職をカミングアウトする発言が、turn には 0 が格納される。type が “talk” または “whisper” の場合は発言またはささやきを示し、agent には発話者のエージェント ID が、idx には発言・ささやきに割り当てられた ID が、text には発言内容が、turn には 1 日に呼び出された発言の回数が格納される。type が “vote” または “attack_vote” の場合は追放投票または襲撃投票を示し、agent には投票対象のエージェント ID、idx には自身のエージェント ID、text には投票

をしたことを示す発言内容, turn は基本的には 0, ただし再投票になった場合は 1 回目の投票の turn=-1 が格納される. type が “execute” または “dead” の場合は処刑または襲撃によって死者が出たことを示し, agent には死者のエージェント ID, text には死者が出たことを示す発言内容, idx と turn には 0 が格納される.

人狼ゲームが開始した時点で, 前述の initialize メソッドが呼び出され, ゲームの初期設定を得ることができる. その後は, update メソッドに diff.data としてゲーム中に発生する情報と, base.info にて基本的なゲームの状況を得ることができる. 人狼推定モデルで必要とする特徴量は diff.data に格納されているものが多いので, update メソッド内で逐次情報を更新し特徴量を作成している. 行動決定モデルには 1 日ごとに更新を要求される物や常に最新の情報を求める物があり, エージェント本体で処理をするものや人狼推定モデルのために作成した情報を整理するメソッドを使用する物など乱立している. ここは改善の余地がある部分である.

次にモデル班の作ったプログラムで使われている特徴量の調査を行った. 調査した時点での特徴量は占い師の CO した順番のリスト (昇順), 霊媒師の CO した順番のリスト (昇順), divine[w/v][agent#] (誰を白とといった回数), day-divine[日付][w/v][agent#]=投票発言回数, suffer[w/v][agent#] (誰が自分を白/黒といった回数) であった.

モデルに入力するデータの前処理の準備が出来たので, 別に新しくクラスを作成して, そこで前処理を行った. そのクラスでは PandasDataFrame 形式の diff.data を Numpy 形式の 304 次元の 1 次元配列のデータに変換する処理を行った. また, 304 次元の 1 次元配列のデータには, 日付, 生存者数, 生存している占い師数, 生存している霊媒師数, 占い師 CO 状況, 霊媒師 CO 状況, 占い結果票, 肯定発言数, 否定発言数, 投票変更数を格納した.

エージェントの行動決定の処理として, 必要な情報を変数としてエージェント内への用意, 騙り, CO 直後の占い結果, 霊媒結果の表明, over する前に 1 度, 推定モデルで算出された人狼たちを怪しむ発言を行う (人間の場合) プログラムの実装を行った. 必要な情報とは, 追放されたエージェントの番号 (1 - 15, 誰も追放されていない場合は 0), 朝襲撃された人の番号 (1 - 15, 誰も襲撃されていない場合は 0), 自分が占い師の場合の占い結果, 自分が霊媒師の場合の霊媒の結果である. また, 騙りの具体的な内容としては, 村人の場合には騙りなし, 人狼・狂人の場合は行動リスト「3」の値によって, 0 なら騙りなし, 1 なら占い師, 2 なら霊媒師に騙り, 占いの対象と結果はでたらめになるようにした.

(※文責: 木村高之)

3.2.2 活動の問題点

後期での活動の問題点としては, 前期の時点で懸念されていた, スケジュールの過密が挙げられる. 後期では 1 週間に 1 度, オフラインでの活動が認められていた. そのため全面オンライン活動だった前期と比べ, 活動がスムーズに進むことでスケジュールの過密が解消されると考えられていたが, 結局のところスケジュールの過密を解消することができず, 様々な問題を引き起こす結果となった. スケジュールの過密によって引き起こされた問題の一つとして, まず予定の変更が挙げられる. 前期の時点では, プロトタイプの作成や, 複数のエージェントの開発・比較を行うことが予定されていた. しかし, スケジュールの過密により, 時間的に複数のエージェントの作成を行うことが困難であったため, 1 つのエージェントの開発のみを行うこととなってしまった. また, 作成した 1 つのエージェントについても, デバック作業やパラメータの微調整, 作成した人狼推定モデ

ル、行動決定モデルの学習など、モデルの性能アップに必要な作業の時間をあまり取ることができなかった。

その他の問題点としては、班分けによる問題点がいくつか挙げられる。後期ではモデル班と設計班に分かれて活動を行った。班分けにより、2つの班がそれぞれエージェント設計作業と、モデル構築作業に専念することができたが、その一方で、もう片方の班のコードの理解が困難になるなど、エージェント開発作業全体の把握をすることが難しくなるという結果になってしまった。また、班で別々のライブラリを使っていることや変数名の統一がされていないことなど、2つの班が並行して無秩序に開発を行ったことで、統合の際に少し手間取る事になってしまった。班分けによる問題点自体も話し合いによって解決できるはずであるから、これについてもスケジュールの過密が起因することが考えられる。

(※文責: 工藤光矢)

3.3 共同開発環境

前期・後期の活動を通して、共同開発環境の構築とツールの使いかたの習得を行った。

(※文責: 松田顕)

3.3.1 Git

Git は分散型バージョン管理システムであり、共同開発に限らず開発では必須となるシステムである。Git はファイルの更新を記録する。これによってなぜこのような実装になったのか、いつだれが変更したのかなどが明らかとなる。また、過去の状態に戻ることができるため、加えた過去の変更をやり直すなど、開発において発生する問題について対処することができる。また、「ブランチを切る」ことでソースコードが複製され、同時に複数の変更を加えることが可能となる点は共同開発において大きな利点となる。グループメンバー全員が Windows 環境を使用していたため、このシステムを利用するために Git bash for Windows を導入した。これにより、Windows 環境でも Unix シェルひいては Git を利用できるようになる。

(※文責: 松田顕)

3.3.2 GitHub

GitHub は、ソフトウェア開発のプラットフォームであり、前述した Git をインターネットを介して複数人で利用できるようにしたサービスである。本年度は基本的にオンラインでの活動となったため、このようなオンラインで開発が行えるようなサービスの利用は必須であった。プロジェクトと、グループ用のリポジトリを作成し、このリポジトリを更新していくことで開発を進めた。

(※文責: 松田顕)

3.3.3 Anaconda3

後期には Python を用いて人狼知能エージェントを作成することが決定した。環境の違いによる問題が発生しないように、Python, そして使用するライブラリはグループメンバーが全員同じ環境で使用するべきである。Anaconda3 はその問題を解決する。Anaconda3 は Python のディストリビューションであり、Python の導入はもちろん、ライブラリについてもバージョンを指定してでの導入が可能である。人狼知能プロジェクトが公開している Python 版の基本エージェントには、バージョンの古いライブラリが存在し、バージョンの指定が可能な点は非常に有用であった。使用するライブラリなどは、それらを環境のセットとして設定でき自由にかつ簡単に切り替えることができるのも Anaconda3 の強みである。これを用いて Numpy, pandas, Keras などの導入を行った。

(※文責: 松田顕)

第 4 章 人狼知能の詳細

4.1 人狼知能プロジェクトについて

人狼知能プロジェクト [1] とは，不確定情報コミュニケーションゲームである「汝は人狼なりや？（通称，人狼ゲーム）」を機械学習を用いて，囲碁やオセロと同等に人工知能による制御を目指す研究分野である．また，このプロジェクトの究極的な目標は「人間と自然なコミュニケーションをとりながら人狼をプレイできるエージェントの構築」である．この目標の実現のために，エージェント技術，人工知能，自然言語処理，HAI（エージェントのインターフェイスの実装）といった技術が必要となる．

（※文責: 木村高之）

4.2 人狼知能大会について

人狼知能大会とは，人狼知能プロジェクト [1] が人狼ゲームをプレイするうえで求められている人工知能技術の発展を目的として，夏と 3 月ごろに開催されている競技会のことである．人狼知能大会には，会話内容を機械処理が容易なプロトコルを用いるプロトコル部門と，会話内容を自然言語で行う自然言語部門の 2 種類がある．グループ内でどちらの部門に出場するのか協議した結果，自然言語部門は自然言語処理を行う必要があり，時間的に厳しいと判断した．従って，本グループはプロトコル部門に出場する予定である．

（※文責: 木村高之）

4.3 プロトコル部門について

プロトコル部門とは，人狼知能プロトコルによるコミュニケーションのみを許可した人狼知能による大会である．人狼知能プロトコルとは，人狼ゲームでの発言を「COMINGOUT Agent[15] SEER」（エージェント 15(自分) が占い師であるとカミングアウトします）のようにプロトコルという形として決まった発言のみを許可することで，自然言語処理ができなくても人狼知能を作成できるフォーマットのことである．また，人狼知能プロトコルを処理するライブラリは Java, NET, Python などの言語で利用が可能である．

（※文責: 木村高之）

4.4 人狼ゲームの詳細

4.4.1 人狼ゲームの概要

人狼ゲームとは、村陣営と人狼陣営の2つに分かれたプレイヤーたちが、各々の勝利を目指して戦うゲームである。ゲームは昼のターンの夜のターンを繰り返すターン制で行われる。昼ターンには全員で議論を行い、投票によって怪しいプレイヤーを一人選出し、選ばれたプレイヤーを追放する。夜ターンには人狼が人間を襲撃したり、特殊能力を持った役職のプレイヤーが能力を使ったりする。これをどちらかの陣営が勝利条件を満たすまで繰り返して行う。村陣営の勝利条件はすべての人狼を追放（処刑と表現したりもする）することであり、人狼陣営の勝利条件は人狼の数と人間の数が同数になることである。

(※文責: 小田中嵐)

4.4.2 役職

各プレイヤーはゲーム開始時にいずれかの役職を与えられる。それぞれについて述べる。

村陣営の役職

村人 特殊能力を持たないため、会話のみをヒントにして人狼を探し出し追放していく。基本的に人狼ゲームにおいて最も人数の多い役職である。

占い師 夜のターンで1日1人を指定して、そのプレイヤーが人狼であるかどうかを調べることができる。人狼陣営のプレイヤーは「自分が占い師である」と嘘をつくことが多いため、本物の占い師として村人陣営から信頼を得ることができるように、高度なプレイが必要である。

霊媒師 夜のターンで前日追放されたプレイヤーが人狼であったかどうかを調べることができる。

狩人 夜のターンにプレイヤー1人を指定して、人狼の襲撃から護衛することができる。護衛されたプレイヤーが襲撃された場合、襲撃は失敗し、そのプレイヤーは生き延びることができる。

人狼陣営の役職

人狼 夜のターンで他のプレイヤーを襲撃する権利を持つ。占い師などの重要な役職を上手く襲撃することが重要である。人狼が全て追放されると人狼陣営の敗北となるので、「自分は占い師である」のような嘘を言うなどして、正体がばれないように動く必要がある。人狼は互いに誰が人狼であるかが分かっているため、人狼同士で協力も必要である。

狂人 特殊能力を持たない人狼陣営の役職である。誰が人狼であるかを推測し、人狼を手助けするように振る舞う必要がある。人狼陣営ではあるが、占い師や霊媒師には「人間」と判定される。狂人自身が死んでいても人狼陣営が勝利すれば勝利判定になる。

人狼ゲームには様々なローカルルールが存在し、役職の名前や能力が多少異なる場合や、その他の役職が追加されている場合があるが、本稿では上記の通り統一することとする。

なお、人狼知能大会では、5 人人狼と 15 人人狼が採用されており、配役は表 4.1 のようになっている。

(※文責: 小田中嵐)

表 4.1 人狼知能大会の配役

役職	5 人人狼	15 人人狼
村人	2 人	8 人
占い師	1 人	1 人
霊媒師	0 人	1 人
狩人	0 人	1 人
人狼	1 人	3 人
狂人	1 人	1 人
合計	5 人	15 人

4.4.3 ゲームの流れ

人狼ゲームは以下のような流れで進行していく。

0 日目

各プレイヤーに役職が振り分けられる。占い師のみが能力を使用することができる。

1 日目昼

ゲームから追放するプレイヤーを議論によって決める。自分の役職をカミングアウト（CO とも言う）したり前日に使用した能力の結果を発言したりすることで人狼だと思われる怪しい人物を投票によって 1 人選び、そのプレイヤーを追放する。実際の画面では図 4.1 のように進行していく。

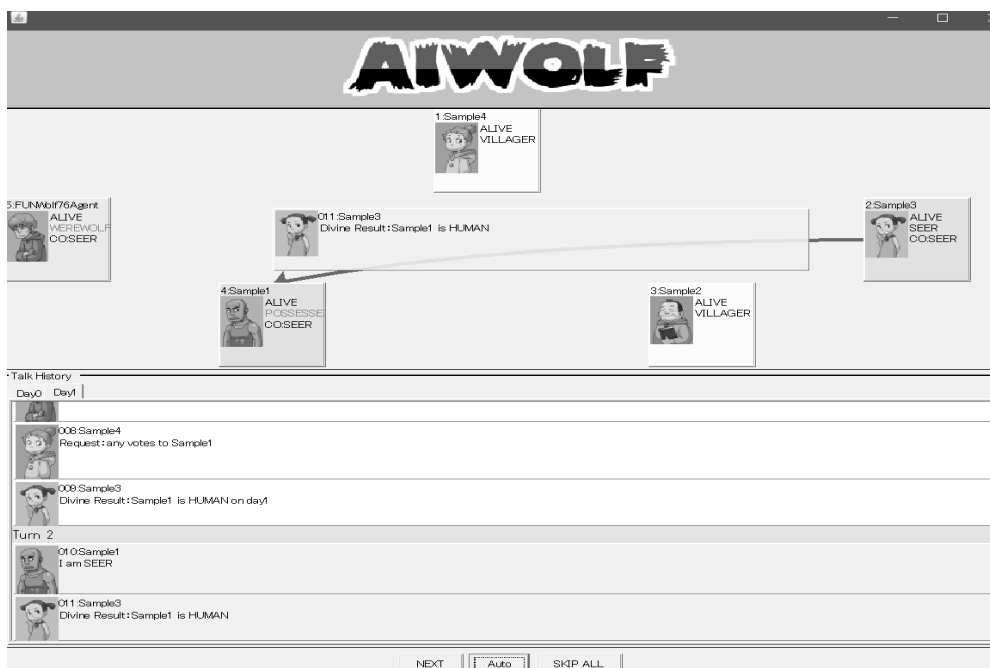


図 4.1 ゲームの進行画面

1 日目夜

特殊能力を持ったプレイヤーが能力を使用する。15 人狼では人狼同士で会話をする
ことができ、襲撃先を相談することができる。人狼の襲撃先が狩人の護衛先と被らなかつた場合、襲撃は
成功し、襲撃されたプレイヤーは次の日の昼ターンの初めに追放扱いとなる。

2 日目以降

1 日目と同じようにゲームを進めていく。これを村陣営か人狼陣営のどちらかが勝利条件を満
たすまで続けていく。

ゲーム中の発言内容や投票先、占い結果などはすべてログデータとして図 4.2 のような形式で保
存される。本グループではこのログデータをもとにニューラルネットワークのモデルを構築した。

```
0,status,1,VILLAGER,ALIVE,cndI
0,status,2,WEREWOLF,ALIVE,Romanesco
0,status,3,POSSESSED,ALIVE,Udon
0,status,4,SEER,ALIVE,LittleGirl
0,status,5,VILLAGER,ALIVE,spicy2
0,divine,4,5,HUMAN
1,status,1,VILLAGER,ALIVE,cndI
1,status,2,WEREWOLF,ALIVE,Romanesco
1,status,3,POSSESSED,ALIVE,Udon
1,status,4,SEER,ALIVE,LittleGirl
1,status,5,VILLAGER,ALIVE,spicy2
1,talk,0,0,5,VOTE Agent[01]
1,talk,1,0,4,VOTE Agent[03]
1,talk,2,0,3,COMINGOUT Agent[03] SEER
1,talk,3,0,1,VOTE Agent[03]
1,talk,4,0,2,VOTE Agent[01]
1,talk,5,1,5,Skip
1,talk,6,1,4,VOTE Agent[03]
1,talk,7,1,3,DIVINED Agent[04] WEREWOLF
1,talk,8,1,2,Skip
1,talk,9,1,1,VOTE Agent[05]
1,talk,10,2,3,VOTE Agent[04]
1,talk,11,2,1,VOTE Agent[04]
1,talk,12,2,4,Over
1,talk,13,2,2,VOTE Agent[04]
1,talk,14,2,5,VOTE Agent[04]
1,talk,15,3,3,Over
1,talk,16,3,2,Skip
1,talk,17,3,1,Skip
1,talk,18,3,4,Over
1,talk,19,3,5,Skip
1,talk,20,4,4,VOTE Agent[01]
1,talk,21,4,2,Skip
```

図 4.2 ログデータ

(※文責: 西村和馬)

第 5 章 エージェントの概要

5.1 概要

我々は「人狼をプレイできるエージェント」を既存のエージェントとは違ったアプローチから構築することを目的としている。人狼をプレイできるエージェントの前例としては、人狼知能プロジェクトが主催する人狼知能大会に参加したエージェントが挙げられる。それら既存のエージェントは役職推定と行動決定の2つの構成に分けられており、役職推定は機械学習、行動決定はルールベースで行うことが主流な方法である。したがって、目的を達成するためには、このような従来の方法とは違った方法を検討する必要がある。グループで話し合った結果、役職推定から行動決定までを機械学習で行うこととし、それを行うための機械学習のモデルにはニューラルネットワークを用いることが最適であると判断した。前期でのモデルのアイデアとしては、全てを1つのモデルで行う、自分の役職ごとにモデルを作成する、他プレイヤーごとにモデルを用いる、といったものが検討された。

後期では、前期のアイデアについてさらに検討した結果、役職推定から行動決定までをニューラルネットワークで行うというアイデアは、ニューラルネットワークが分類問題と回帰問題を扱うという性質上、行動決定を行うのが困難であるという結論に至ったため、役職推定までをニューラルネットワークで行い、行動決定を強化学習手法の一つである、DQN(Deep Q-Network)で行うというアイデアが提案された。後期で新しく提案された手法についても、既存のエージェントとは違った手法であるといえるため、役職推定にニューラルネットワーク、行動決定にDQNを用いて、Python言語で開発を行っていくことに決定した。

活動の結果完成したエージェントは「FUN Wolf」と命名され、最終的に図 5.1 で示されたような概形となった。本章を通して構造および機能の詳細を述べていく。

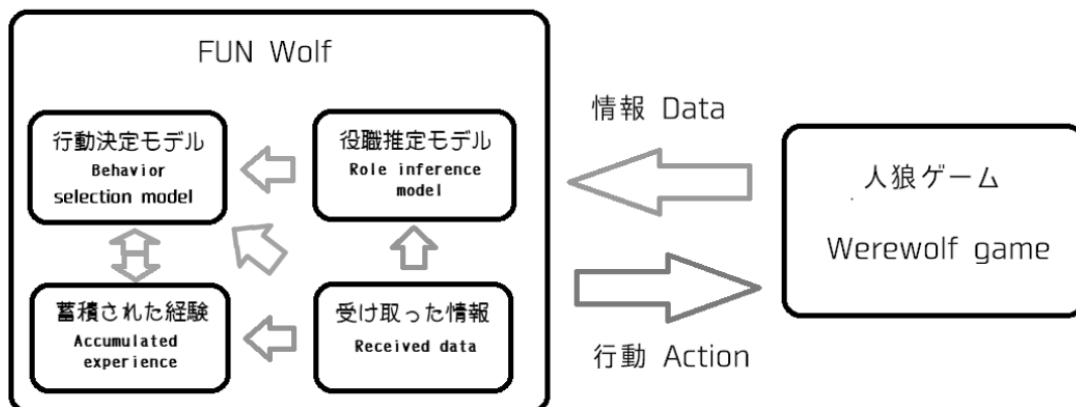


図 5.1 エージェント「FUN Wolf」の概形

(※文責: 工藤光矢)

5.2 ニューラルネットワークについて

本節では、本年度のプロジェクト活動で使用した技術や用語について解説を行う。

ニューラルネットワーク (Neural Network) とは、脳内で形成されるニューロンのネットワークを模した数学モデルである。ニューロンとは脳を構成する細胞であり、入力部である樹状突起、出力部であるシナプス、伝送部である軸索によって構成される。このニューロンは樹状突起から電気信号を入力として受け取り、その電位が一定値を超えると軸索を通して出力としてシナプスを介して次のニューロンに電気信号を送る。このニューロンのふるまいを数学モデルとして表したものの一つにコネクショニストモデルがある。このモデルは非常に簡略化されたものであるが、これはあくまでもニューラルネットワークを形成する要素の一つであるため、一般的にニューロンの数学モデルとして使用される。コネクショニストモデルは以下のようにニューロンのふるまいを表現している。ただし、 y は出力、 x_i は第 i 番目 ($i = 1, 2, \dots, n$) の入力、 w_i は x_i に対するシナプス結合の強さ、 θ は閾値、 f は活性化関数や伝達関数と呼ばれ、階段関数やシグモイド関数などの非線形関数が採用される。

$$y = f\left(\sum_{i=1}^n w_i x_i - \theta\right)$$

以上で説明したニューロンが多数結合しあって電気信号のネットワークを形成するのがニューラルネットワークである。数学モデルとしてのニューロンを結合する際は、層状ネットワーク、相互結合ネットワークなどその結合形態によって種類が存在する。

このニューラルネットワークは機械学習の分野で活用されており、ディープラーニングなど最新の研究分野でも応用されている。ディープラーニングはニューラルネットワークにおいて、ネットワークを深くしたものであり、具体的には入力層と出力層を除いた隠れ層と呼ばれる層を2層以上にしたものを指す。

ニューラルネットワークの学習とは、 w_i, θ のパラメータを調節し理想的な出力を実現させることである。パラメータの調節には、教師データと呼ばれる理想的な入力と出力の集合を用いて行う。学習には様々なアルゴリズムや手法が存在する。このようなアルゴリズムは最適化アルゴリズムやオプティマイザ (Optimizer) と呼ばれる。学習にはアルゴリズム以外にも活性化関数の選択も大きく関わってくる。

損失関数とは、実際に出力された値と期待される出力の値の誤差を示す関数である。この値が小さければ小さいほどそのモデルは優秀であるといえるが、過学習についても考慮する必要がある。

過学習とは、そのモデルが教師データに過剰に適合しすぎた状態である。これにより教師データに対しては正確な回答を出すことはできるが、未知のデータに対して正答率が大きく悪くなる。過学習が発生する原因はいくつかあり、その対策も複数存在する。本年度の活動で使用した過学習の対策はドロップアウトである。ドロップアウトでは、ある更新についてモデルの層ごとにノードのいくつかをないものとして学習を進める。これにより各パラメータの更新が制限され、結果として過学習を防ぐことができる。

(※文責: 松田顕)

5.3 エージェントの詳細

本節では、どのようにして機械学習モデルを人狼知能エージェント内で運用しているかについて解説を行う。図 5.2 におおまかな処理の流れを示す。

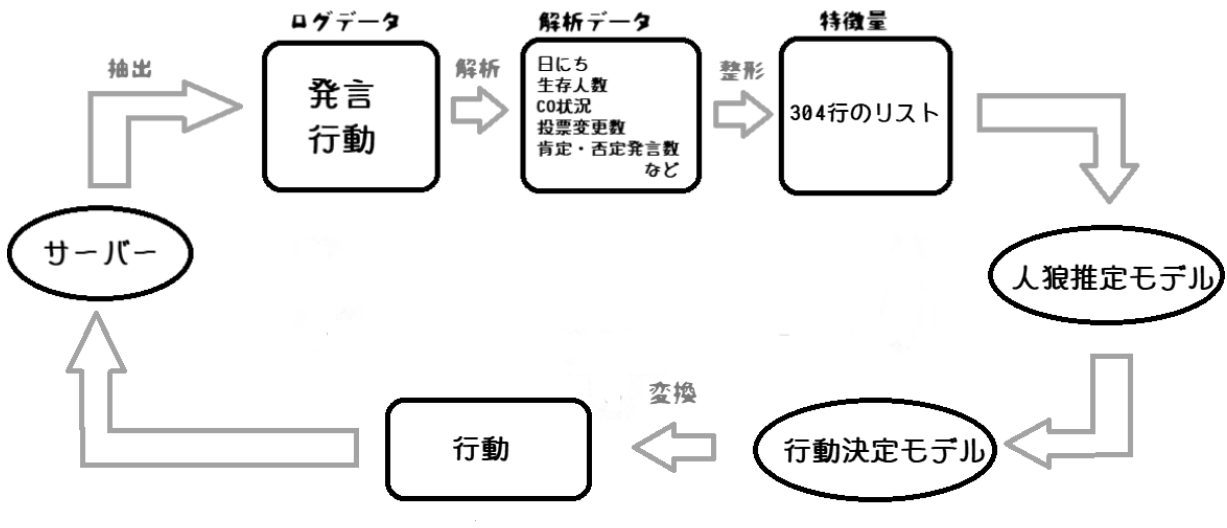


図 5.2 処理遷移図

エージェントは公式が用意している Python 用プラットフォームの AIWolfPy[8] を元に構成されている。AIWolfPy は、Java で動作している人狼知能サーバーとの接続や、データ形式の変換を担っている。詳しくは 3.2.1 の設計班で解説したとおりである。新たに追加したクラスとして、feature_maker がある。feature_maker は各モデルが要求する特徴量を求める。役職推定モデルのために特徴量を作成する際は、基本的に diff_data を update メソッドで人狼知能サーバーから受け取り、同じメソッド内で feature_maker のセッターに引数として渡すことで特徴量の作成を行っている。セッターでは与えられた情報を日にちごとに、情報が公開された順番で蓄積するような形で特徴量を作成するようにしている。セッター内ではまず Pandas の DataFrame 形式で多次元的に格納しているが、最終的には 1 日を 1 列の Numpy の ndarray 形式に変形している。その特徴量は update メソッド内でゲッターを通して呼び出され、その直後に役職推定モデルに渡されている。

役職推定モデルの結果はそのまま行動決定モデルの特徴量として使用される。役職推定モデルは強化学習を使用しており、最新の情報が求められる場合がある。そのため update メソッド内で逐次更新する処理を追加している場合がある。これに関しては、より分かりやすく効率的な実装が考慮できる部分である。そのほかに関しては、新たに feature_maker クラスに新たなセッターとゲッターを作成している。役職推定モデルに作成した特徴量の一部を取り出している特徴量も存在する。行動決定モデルはその出力として投票対象や行動基準を返す。エージェント内ではその出力に合った行動を行うように各行動メソッド内で調整している。行動メソッドは、もともと AIWolfPy に用意されているもので、talk, vote, attack, divine, guard など、人狼ゲーム内で想定できるすべての行動に対して用意されている。自分の役職でとれる行動メソッドが人狼知能サーバーによって呼び出されて、正しい戻り値を返すことで行動が成立する。

(※文責: 松田顕)

5.4 機械学習モデルの詳細

5.4.1 開発環境

本グループで人狼知能エージェントの開発を行うに際して、Python のライブラリ・環境を統合管理するツールである Anaconda3 を用いた。以下はその主要な構成である。

- Keras (version 2.3.1)
- TensorFlow (version 2.1.0)
- NumPy (version 1.19.1)
- pandas (version 1.1.3)
- 人狼知能プラットフォーム (version 0.6.2)
- AIWolfPy (version 0.4.9)

Keras はニューラルネットワークモデルの構築を比較的容易にするライブラリであり、本グループにおいて、主要なモデル構築手段となった。

TensorFlow は Python で機械学習を行うためのライブラリ群であり、Keras はこのライブラリに強く依存しているため、導入が必要である。

NumPy は主に行列等における数値データを高速に処理するための計算支援ライブラリである。

pandas はデータ解析を支援するライブラリであり、後述の人狼知能プラットフォーム及び AIWolfPy との連携に利用する。

人狼知能プラットフォームは、人狼知能プロジェクト公式サイトで公開されているエージェント開発を支援する基盤となるソフトウェアである。サンプルプレイヤーのクラスが入ったファイルや人狼知能ゲームサーバを実行するファイル、ゲームの設定ファイルなどが含まれており、これらはすべて Java で開発されている。

AIWolfPy は Java プログラムである人狼知能プラットフォームに、Python プログラムから接続するための橋渡しとなるソフトウェアである。

(※文責: 西村和馬)

5.4.2 人狼推定モデル

人狼エージェントは、役職推定と行動決定の 2 つの構成に分けられている。我々は役職推定の部分において、占い結果やプレイヤーの発言などの試合の情報から、誰が人狼であるかを推定する人狼推定モデルを構築した。本項では、人狼推定モデルを構築するまでの手順を述べる。

まず、人狼推定モデルを構築するに当たり、ログデータの読み出しを行った。読み出しに用いたデータは、第 4 回人狼知能大会のログデータ (CEDEC2018) である。第 4 回人狼知能大会のログデータは、5 人狼と 15 人狼の試合データが 100 試合ごとに 204 のフォルダで分けられており、それらログデータをプログラム上で扱えるように、Python の文字列リストに読み出しを行った。

次に、読みだしたログデータから特徴量の抽出を行った。特徴量とは特徴を数値化したものを指し、教師ありの機械学習のモデルを構築するためには膨大な量の特徴量を学習させる必要がある。今回の人狼推定モデルで言えば特徴量とは、日付、占い結果、プレイヤー同士の発言など、人狼を推定するのに必要な特徴を数値化したものである。なお特徴量の選定は、先行研究で

ある梶原, 鳥海, 稲葉, 大澤, 片上, 篠田, 松原, 狩野 (2016) および大川, 吉仲, 篠原 (2017) が使用したものを踏まえた上で行われた [9][10]. これらやその他必要そうな特徴量を, Python を用いて試合の日ごとに抽出した. 抽出した特徴量の詳細を表 5.1 に示す. また, データを Keras で構築するニューラルネットワークモデルで扱えるように, 抽出した試合の日ごとの特徴量を 304 次元のデータとなるように要素数 304 の 1 次元 NumPy 配列として整形した.

表 5.1 ログデータから抽出した特徴量

特徴量	説明	データ構造	データ数
日付	今日が何日目であるか	1	1
生存者数	生存しているエージェントの数	1	1
占い師数	生存している占い師の数	1	1
霊媒師数	生存している霊媒師の数	1	1
占い師 CO 状況	どのエージェントがどの順番で占い師 CO をしたかの情報	1×15	15
霊媒師 CO 状況	どのエージェントがどの順番で霊媒師 CO をしたかの情報	1×15	15
占い結果	どの占い師がどのエージェントを占ってどのような結果だったかの情報	15×15	225
肯定発話数	あるエージェントの肯定の発話の回数	1×15	15
否定発話数	あるエージェントの否定の発話の回数	1×15	15
投票変更数	あるエージェントの昼に行った投票宣言と実際の投票先が違っていた回数	1×15	15
		合計	304

ログデータの読み出しと特徴量の抽出を行った後は, 人狼推定モデルの本体の構築を行った. 人狼推定モデルは, 日付, 占い結果, プレイヤー同士の発話など, 人狼を推定するのに必要な特徴から, だれが人狼であるかを推定するニューラルネットワークモデルである. 構築には, Python のニューラルネットワークのライブラリである Keras を用いた. モデルのアーキテクチャ (表 5.2) としては, 入力層と出力層を含めて全 4 層で構成されており, 出力層以外は ReLU 関数により, 0 以下の値を 0, 0 以上の場合は値をそのまま伝達する手法をとっている. また, 過学習を防ぐために中間層ではドロップアウトを適用しており, 訓練時の更新において 0.2 の確率でランダムに入力ユニットを 0 としている. 入力には抽出した 304 次元の特徴量のデータとし, 出力は事前に求めておいた組み合わせ ID (表 5.3) を出力するものとした. 組み合わせ ID とは, 1 人の人狼がいる 5 人人狼と, 3 人の人狼がいる 15 人人狼で考えられる人狼の組み合わせを全て列挙し, 列挙した組み合わせに先頭から順に割り当てた, 0~459 の値である. 例として, 入力した情報からプレイヤー 11, 14, 15 が人狼であると推定できるなら組み合わせ ID の 455 を出力するというものである. このような組み合わせ ID を出力層では SoftMax 関数を用いて出力する. SoftMax 関数とは複数の出力値の合計が 1.0 になるような関数であり, 出力結果がどのような値を取ろうとも 0.0~1.0 の間に正規化を行う役割を持つ. また, 損失関数には cross entropy を用いた. これは出力層の活性化関数が SoftMax 関数である場合に基本的に用いられる. オプティマイザには, RMSprop を用いた.

RMSProp は学習状況によって変化する動的な学習率を持つことで、学習結果の安定化を促進する性質を持つ関数である。

表 5.2 人狼推定モデルアーキテクチャ

	ユニット数	活性化関数
入力層	304	
	1840	ReLU
		Dropout(0.2)
	920	ReLU
		Dropout(0.2)
出力層	460	SoftMax
Optimizer		RMSprop
損失関数		categorical_crossentropy

表 5.3 人狼推定モデルの出力

組み合わせ ID	人狼と推定されるプレイヤー		
0	1		
1	2		
2	3		
3	4		
4	5		
5	1	2	3
6	1	2	4
7	1	2	5
...		...	
455	11	14	15
456	12	13	14
457	12	13	15
458	12	14	15
459	13	14	15

最後に完成した人狼推定モデルに、入力に 304 次元の特徴量のデータ、正解に人狼の組み合わせ ID である学習用データを用いて学習を行い、評価データを用いてモデルの精度を確認することで、人狼推定モデルを構築するまでの全行程が終了となる。

(※文責: 工藤光矢)

5.4.3 行動決定モデル

行動決定モデルを実装するにあたって、Deep Q-Network (DQN) モデルを採用した。DQN モデルは Q-Learning という強化学習の一種にニューラルネットワークの仕組みを導入することで、より複雑な対象に適応させることを可能にした学習モデルである。本実装における具体的なフローを述べる。なお DQN モデルの学習に際し ReNom 社作の理論説明を利用した [11]。

初めにモデルにより決定することのできる行動の種類を設定した (表 5.4)。全 4 種類の行動を定め、それぞれに複数のパターンをとることが可能である。

「役職カミングアウトのタイミング」は占い師や霊媒師が自分の役職を公表する時のタイミングを示す。会話の開始直後、自分以外が自分の役職を宣言した時、自分が人狼であると占われた時、自分が人間であると占われた時、自分に投票が集中している時の 5 パターンに区分される。

「特殊行動の対象」「投票行動の対象」はその行動の対象とするプレイヤーの番号によって表される。自分自身も範囲に含まれてしまうため、後述の報酬設定で調整を施している。

「人狼・狂人の性格」は人狼・狂人になった際、他の役職へのなりすまし (専門用語で騙りと言う) を行うかどうかを決める。騙りをしない、占い師を騙る、霊媒師を騙るの 3 パターンに区分される。騙りをする場合、前述の「役職カミングアウトのタイミング」に従って、本物と同様の手順で行動する。

実際に行われる行動は、[2 5 3 1] のようにパターンの組み合わせによって表現される。

表 5.4 行動パターン

種別	パターン数
役職カミングアウトのタイミング	5
特殊行動の対象	15
投票行動の対象	15
人狼・狂人の性格	3

次に行動・ゲーム状況に対して報酬を導入した。報酬はゲームの有利・不利を数値的に示したものであり、値が大きくなればなるほど有利であり、値が小さくなればなるほど不利であると定めている。有利な行動・ゲーム状況とは主に、特殊行動 (占い・襲撃) を敵対する能力者に対して使用すること、自分の陣営が勝利することを指す。これらの行動・ゲーム状況をとることで正の報酬を得ることができる。反対にそれらの行動・ゲーム状況をとらない場合には値が増加することはなく、値は低いままとなる。また想定されていない行動 (自分自身に特殊行動・投票を行う・すでに襲撃されているプレイヤーを襲撃するなど) に対しては負の報酬を与えるものとした。

DQN モデルの本体は報酬の値をシミュレートする。モデルのアーキテクチャ (表 5.5) は入力層を含め全 7 層で構成されており、出力層以外は ReLU 関数により 0 以下の値を 0、それ以外の値を重みを掛ける操作を除きそのまま伝達する手法をとっている。重みとは与えられた入力値をシミュレートしたい値に修正するための変数のことを指す。入力是人狼知能モデルで利用した特徴

量, 自己プレイヤー情報, 行動パターンの組み合わせ, 人狼推定モデルの出力, これらを結合したものとした. 出力は前述の通り報酬の値のシミュレートであり, 可能な限り報酬の値を模倣することを期待している. ただし, ここでの報酬は前述した報酬値そのままではなく, 後述の変形を行った後の値である. 入力の実行動パターンの組み合わせを複数検証することにより, 最も高い出力値を算出する行動パターンの組み合わせを求める. その組み合わせが最終的な行動予測となる. またモデル内の計算において, 各数値が計算可能範囲を超えないようにするため, 損失関数に Huber 関数を設定した. Huber 関数は出力値と実報酬の値との平均二乗誤差に基づき, 2つの数学的関数を切り替える動作を行う. この動作により, 数値が一定より大きい場合は緩やかな関数, 小さい場合は傾き 1 の一次関数となり, 数値の異常な変化を防ぐことが可能である. 同様に Optimizer として設定した Adam には RMSprop の性質に加えて, 異常値がモデルに与える影響を抑制する効果があり, モデルの一般化に貢献している.

表 5.5 行動決定モデルアーキテクチャ

	ユニット数	活性化関数
入力層	347	
	347	ReLU
	347	ReLU
	347	ReLU
	347	ReLU
	347	ReLU
出力層	1	Linear
Optimizer		Adam
損失関数		Huber

最後に学習及び行動決定を確定する操作を作成した. 学習進行度を示すパラメータとして ε ($0.1 < \varepsilon < 1.0$) を設定し, $(1 - \varepsilon)$ の確率でモデルに基づく行動決定を行う. それ以外の場合はパターン範囲の中からランダムで行動を行う. これは学習の硬直を避けるためであり, 未知の行動を無視しにくくなるのが効果として考えられる.

学習に関して, ゲーム中の情報更新を受け取るたびに現在のゲーム状況 (人狼知能モデルで利用した特徴量, 自己プレイヤー情報, 人狼推定モデルの出力), 現在の報酬, 現在決められている行動の組み合わせ, これらをそれぞれ計算・処理し 1 つの要素にまとめた後, 記憶領域スタックに蓄積する. 最終的に 1 ゲームの終了時, この要素をそれぞれモデルに学習させる操作を行っている. 学習の際, 現在の要素とその次の要素を関連付けた値をラベルとして与えている (最終行動を除く).

$$R = \gamma \times r_{next} + r_{real}$$

ラベル R はモデルに対して与える「正答」であり, この値をモデルはシミュレートする. r_{real} は現在の実報酬値であり, モデルによらない計算した現在の報酬そのままが入力される. r_{next} は次の報酬予測であり, 現在の次の要素に対してモデルを適応した時, もっとも優れた行動の報酬予

測値を代入する（現在の次の要素に格納されている実報酬値は使用しない）。 γ ($\gamma = 0.8$) は割引率であり、未来の行動にどれだけ重きを置くかを定めている（大きいほど重視）。この式により、現在のみにおける優秀行動だけでなく、未来（次の状況・行動）を考慮した優秀行動を選択することが期待できる。

ラベル R および現在のゲーム状況、現在決められている行動の組み合わせ、人狼推定モデルの出力、これらをモデルに学習させ、学習工程は終了となる。

(※文責: 根上樹)

5.5 評価実験

人狼推定モデルと行動決定モデルにおいてそれぞれ学習・運用実験を行った。実験日は 2020 年 12 月 2 日であり、使用モデルの詳細・学習に利用するログデータ等に関しては前述の通りである。また行動決定モデルが人狼指定モデルの出力を参照する都合上、人狼推定モデル、行動決定モデルの順に実験を行った。

人狼推定モデル

学習を 2 つの試行に分けて実施した。第 1 学習においては、学習用データ 8000 ゲーム、検証用データ 2000 ゲームについて 100 回学習を行わせた。第 2 学習においては、学習用データ 8000 ゲーム、検証用データ 2300 ゲームについて 100 回学習を行わせた。

結果として、学習用データに対して 75%、評価用データに対して 63% の精度で正しい人狼の組み合わせを算出することができた。正しい人狼の組み合わせは全 460 通りのうち 1 つのみであり、ランダムな選択に対しては明確な差があると考えられる。

行動決定モデル

5 人プレイヤーによる人狼 1500 ゲーム、15 人プレイヤーによる人狼 150 ゲームについて 1 回ずつ学習を行わせた。

学習後、5 人プレイヤーによる人狼 30 ゲーム、15 人プレイヤーによる人狼 30 ゲームを合計した 60 ゲームについての勝率を算出した結果、43.3% となり、ランダムでの同試行回数における結果、43% と有意な差は無かった。また行動パターンが少ない種類に収束してしまっていることが確認できた。

(※文責: 根上樹)

5.6 考察

今回、機械学習モデルを用いて人狼の推定と行動の決定を行うエージェントの作成を行った。

人狼の推定は、ランダムに人狼を推定した場合と比較すると、明確な差があり、ランダムに人狼を推定するエージェントよりも、高い精度での推定を行うエージェントが完成したといえる。しかし、学習回数を増加させるなどの細かな調整を行うことや、人狼推定モデルに入力する特徴量を増やすことなど、改変を加えることにより、精度をさらに向上させることが可能であると考えられる。

一方で、行動の決定については、ランダムで行動を決定するエージェントと勝率を比較すると、有意な差は無かった。この原因として、行動決定モデルに入力する特徴量が不足していたこと、モデルの学習回数が不足していたことが挙げられる。また、行動パターンが、「エージェント A が生存している限り、そのエージェントに毎回投票する」といった具合に、少ないパターンに収束してしまった。この原因は、行動とそれに対する報酬の多様化にあると考えられる。これらの原因から、行動決定モデルの改善案として、モデルの根本的な設計を見直すこと、モデルの学習回数を増加させること、毎回同じ対象に同じ行動をしていることに対して負の報酬を与える、というように報酬を見直すことが挙げられる。

(※文責: 西村和馬)

第 6 章 活動結果

6.1 詳細な活動結果

6.1.1 前期

前期の活動として、後期活動の開始と共にエージェントの開発が進められるように開発環境を整え、人狼知能の概要を学び、理解を深めることとした。中間発表では、現時点で学んだ知識をまとめ、それを発表した。具体的な活動として、まずは輪読を行った。人狼知能の専門書 [3] を分担して読み、人狼知能がどのように構成されるかを学んだ。基本的なルールや初期設定などからそれぞれの役職ごとの基本的な動きまで、人狼知能の基礎を理解することができた。また、機械学習の基本についても輪読を行い、実際に libSVM を用い機械学習を活用した人狼知能エージェントの例を学んだ。機械学習に関しては、今後も知識を深めていくために、継続して輪読を続けていく必要がある。

また、2020 年 7 月 5 日に行われたオンラインで開催された人狼知能セミナー 2020 に参加した。ここでは基本的な人狼知能エージェントの作成方法について学ぶことができた。セミナーの内容に関しては、すでに輪読で扱った内容と被る部分も多かったが、これまで Java での開発を予定していたエージェントの Python を用いた開発方法を学ぶことができた。これにより、Java よりも機械学習ライブラリの豊富な Python で開発を進めていく方針に変更した。

反省点として挙げられる点は、開発環境の構築に時間がかかった点である。特に、今年度のプロジェクトはオンラインで行われたこともあり、それぞれが開発環境を整える必要があったが、音声と画面共有だけで情報を共有する必要があり、想定以上に時間がかかってしまった。前期では Git, GitHub, Java, Python, libSVM を利用したが、いずれも導入に時間がかかり、作業が手間取ってしまった。後期は対面の機会が増えるため、前期の反省点を活かし、導入をスムーズに行う必要がある。

(※文責: 小田中嵐)

6.1.2 後期

後期の活動として、人狼の推定と行動の決定を、機械学習によって実現させるエージェントの開発を行った。成果発表会では、完成したモデルの構築の手順や、エージェントの動作を発表した。具体的な活動として、モデル班と設計班に分かれて活動を行った。モデル班は、ニューラルネットワークを用いて人狼を推測する、人狼推定モデルの制作と、人狼推定モデルの出力や経験、ゲームの情報から DQN(Deep Q-Network) を用いて最適な行動を決定する、行動決定モデルの制作を行った。設計班は、モデルに適切な特徴量を渡し、そのモデルの出力から行動などの処理を行うプログラムの構築を行った。モデルの制作とプログラムの構築が完了したのち、それぞれの班の制作物を統合し、エージェントが完成した。その後、完成したエージェントに対して学習・運用実験を行った。今年度で開催された人狼知能大会は 1 回のみであり、エージェントの提出期限が 8 月と早かったため、人狼知能大会へ出場することはできなかった。

反省点として挙げられる点は、スケジュールの過密である。1週間に1度、対面で活動することにより、スムーズに活動が進み、スケジュールの過密が解消されるかと思われたが、解消されなかった。結果として、前期終了時に予定していた、複数の形式のエージェントを制作し、比較することができず、1つのエージェントのみの制作に留まってしまった。そのうえ、モデルの性能アップができなかったことや、話し合う時間が十分に取れず、もう片方の班のコードの理解が困難になる等により、エージェント開発作業全体の把握が難しくなった、といった更なる問題を引き起こすことになってしまった。

(※文責: 西村和馬)

6.2 中間発表

6.2.1 中間発表の準備

中間発表の準備として、ポスターとウェブサイト、発表用スライドを作成した。今年度のプロジェクト中間発表はプロジェクト紹介動画かウェブサイトの選択であったが、本グループではウェブサイトを選択した。ポスターはグループBとグループCと共同で作成し、1枚にまとめた。ウェブサイトはGoogleサイトで作成した。人狼知能を知らない人が見ても理解できるように、人狼の背景からルールを説明し、文章だけでは伝わりにくいエージェントは図を用いて表現した。発表用スライドは、質疑応答の際に使用するために作成した。担当教員からは曖昧な表現や伝わりにくい言い回しについて修正のアドバイスをいただいた。

今年度の中間発表は、前後半に分かれ、さらにその中で3回に分けた1回15分の質疑応答を、参加者を入れ替えながら行われた。参加者は事前に動画またはウェブサイトを参照したうえで質問をする形式であったが、数が多く内容が混乱する恐れがあったため、当日は初めの5分程度で本プロジェクトおよび各グループの概要を説明した後、質疑応答を行うスタイルに進めた。そのため、スライドを用いた質疑応答のリハーサルを行い、時間内に発表できるよう原稿を簡潔にまとめ、他グループへ実際に想定される質問をすることで受け答えの練習を行った。また、当日に円滑に進められるよう、Zoomの操作についての確認を念入りに行った。

(※文責: 小田中嵐)

6.2.2 評価結果

中間発表は2020年7月17日にオンライン上で行われた。

発表後、参加者にGoogleフォームで発表内容の評価をしてもらった。評価項目は、全体の発表技術と各グループの発表内容についてである。それぞれ1点から10点までの10段階で点数をつけてもらい、その理由やアドバイスを記入してもらった。

中間発表で得られた評価点とコメントを以下に記す。

- ウェブサイトが見やすくわかりやすかった
- 動画がないと活動内容がわかりにくい
- 共有された画面が見にくかった
- 質問者がいない場合に補足説明を行い、素早く対応できていた

- 質問にしっかりと答えられていた
- 進行がスムーズでよかった
- 活動内容がわかりにくかった
- 人狼知能大会についての説明がほしかった
- 目標が明確に設定されているのがよい
- 着眼点がおもしろい

表 6.1 中間発表の評価

評価	発表技術	発表内容
10	5	5
9	12	10
8	11	12
7	6	8
6	2	3
5	4	2
4	0	1
3	0	0
2	0	0
1	0	0
平均	8	7.9
合計	40	41

(※文責: 小田中嵐)

6.2.3 評価の考察

表 6.1 より平均点は発表技術が 8 点、発表内容が 7.9 点であった。コメントにもある通り、質疑応答の進行と受け答え自体は入念にリハーサルを行った成果が出ていると考えられる。また、ウェブサイトもわかりやすかったというコメントがあり、発表自体は成功であったと言えるだろう。しかし、反省点として、画面共有がうまくいかなかった点、動画がないため具体的なイメージがしにくかった点、人狼知能大会の説明が詳しくされていなかった点、活動内容がわかりにくかった点などが挙げられた。前半の 2 点に関しては今後オンラインでの発表がある際に検討したい。後半の 2 点に関しては、期末発表ではわかりやすく伝えられるよう意識したい。

(※文責: 小田中嵐)

6.3 最終成果発表

6.3.1 最終成果発表の準備

最終成果発表の準備として、ポスター、ウェブサイト、発表用のスライドを作成した。発表方法はプロジェクト全体でスケジュールが遅れ気味であることを踏まえ、中間発表で作成したひな形をそのまま流用できるウェブサイトを選択した。全体的な構成を見直し、より分かりやすい説明ができるように工夫した。具体的には、図や表を用いた可視的な説明を追加し、ポスターでは文量を減らすことで最も重要な点が伝わるように作成した。ウェブサイトでは、詳細なモデル構築の手順やエージェントの動きなどについて説明し、ポスターだけで説明しきれない内容を補う形にした。

最終成果発表は中間発表と同様にオンライン上で開催された。中間発表ではプロジェクトごとにミーティングを開き、プロジェクト間の移動の際にはその都度ミーティングを入退室する必要があるが、今回はプロジェクト学習 WG が開くミーティングルームの中に各プロジェクトのブレイクアウトルームが割り当てられる形式に変更された。発表形式については中間発表と同じであった。発表に向けてグループ B とグループ C と合同でリハーサルを行い、口頭発表とそれぞれのグループに対しての質疑応答を行うことで、本番を想定したリハーサルを行うことができた。リハーサルを通して、他のグループから得られた質問を参考に、ウェブサイトに新たに説明を加え、一部曖昧だった表現の修正を行った。また、質疑応答の際、質問がなかった場合に司会から追加の説明を促すような質問を用意し、沈黙の時間が続かないような注意を心掛けた。全体を通して、限られた時間の中で念入りに準備ができたと感じている。

(※文責: 小田中嵐)

6.3.2 評価結果

最終成果発表は 2020 年 12 月 4 日にオンライン上で行われた。

発表後、参加者に Google フォームで発表内容の評価をしてもらった。評価項目は、全体の発表技術と発表内容についてである。それぞれ 1 点から 10 点までの 10 段階で点数をつけてもらい、その理由やアドバイスを記入してもらった。最終成果発表で得られた評価点とコメントを以下に示す。

- 活動内容と成果がわかりやすく伝わってきた
- ウェブサイトのデザインをもう少し工夫したほうがよかったと思う
- 司会、質疑応答がはっきりとしていてよかった
- 手短な説明がありがたかった
- 説明が簡潔でわかりやすかった
- 質問のない時間がないように工夫されていた
- 進行や発表が聞き取りやすかった
- 実際に動くデモ動画の紹介などがあると成果物の雰囲気がつかみやすかった
- 質問しやすい環境が整えられていると思った
- 一目でイメージがつくようにもっとイラストや写真が多くてもよかったと思う
- 説明は丁寧であったが、もう少し内容をまとめるとよいと思った

- ウェブサイトのデザインを統一したほうがよい
- 専門用語の解説がほしかった
- 用いた AI の設定や、実験設定など、表にして提示してあると分かりやすかった
- 深層学習に対する詳細な質問にも回答できており、しっかりと勉強していることが良くわかった

表 6.2 最終成果発表の評価

評価	発表技術	発表内容
10	3	3
9	8	9
8	8	9
7	9	11
6	4	3
5	1	1
4	0	0
3	0	0
2	0	0
1	0	0
平均	7.8	7.9
合計	36	36

(※文責: 小田中嵐)

6.3.3 評価の考察

表より平均点は発表技術が 7.8 点、発表内容が 7.9 点であった。いずれも中間発表より点数を伸ばすことができなかった。その原因として、特にウェブサイトの問題があったと考えられる。ウェブサイトはポスターや口頭ではカバーしきれない詳細な情報をまとめているが、本プロジェクトが 3 グループあることや資料を閲覧できる時間が限られていることなどを考慮するのであれば、もう少し量を控え、さらに図や表を増やして簡潔にまとめるべきだったのかもしれない。専門用語の解説やデモの動画がほしいというコメントが多く見られるのも、我々はわかっているつもりで説明を進めている内容が、実際はうまく伝わっていない、伝わりにくい内容であったためであるように感じられる。つまり、読み手の配慮がされていなかった点がこのような結果を引き起こす原因になったと考えられる。文章での説明を必要最小限の範囲に止め、デモ動画やイラストなどで成果物が実際に動いている様子を示すことで、一目見てイメージできるような工夫が必要であったと考える。しかし、開発スケジュールの遅れを踏まえ成果物を形にすることを優先した結果、動画を作るまで手が回らなかったのは反省すべき点である。一方で、中間発表での反省点であった活動内容のわかりにくさについては改善できた。また、発表に関してもリハーサルを重ねた成果がうまく発揮できた。特に、質問が出なかったときに司会から補足の説明を促すように各グループに対して質問を行ったことにより、空白の時間が無くなったうえに追加の情報を提示することもでき、質疑応答がより活発な時間となった。これについてはコメントにもあるように好評であったため、成功で

AI Love Deep Learning

あったと言える。総じて、前期の反省活かしきれない部分もあったが、反省を活かし改善できた点もあった。

(※文責: 小田中嵐)

第 7 章 個人活動

7.1 松田顕

7.1.1 前期

私はプロジェクトリーダーを務め、プロジェクト全体のまとめを行ってきた。中間発表などのプロジェクト全体として活動するための最低限の役割は果たしたと自負している。グループでは、特に共同開発環境の構築に貢献した。私は Git, GitHub の使用経験があり、グループ活動の序盤では Git の導入から使いかたの解説を行い、GitHub のリポジトリの作成と実践までを担当した。使用言語を Python に決定した際には、Python のディストリビューションである Anaconda の導入と使いかたの解説も行った。活動中には積極的に意見を出すようにして、例えば使用する機械学習モデルで話し合いが行き詰ったときは取得可能な入力情報、欲しい出力情報からニューラルネットの使用を提案した。人狼知能セミナーでは Python のコースに参加し、そこで学んだことを共有した。その結果人狼知能エージェントの作成は Python で行うことが決定された。中間発表ではポスターの作成を担当した。文章の作成や得意の英語を生かした英文の作成に尽力した。中間発表当日には質疑応答を務めた。人狼知能についての調査も個人的に行い、論文を読みまとめたものを共有するなど活動に必要な情報を収集している。

(※文責: 松田顕)

7.1.2 後期

後期は前期と同じようにプロジェクトリーダーとしての活動を続けた。特に後期はグループごとに対面での活動が開始されたため、どのグループが登校するかといった管理も行った。グループで学習を行う際には Pandas や Numpy の資料を提供した。Colaboratory を使用した経験から、学習や開発においてそれを使用する提案を行った。これらにより効率的な学習と実践への適用を促すことができたと自負している。グループでは設計班に所属して開発を行った。開発中には前期にも行った Git, GitHub の使いかたを教え、結果班員がこれらの使いかたを習得していった。開発ではエージェントを効率的かつ正確に動作するように心がけた。これを実現するために、モデル班とのコミュニケーションを増やし、要求されている特徴量を逐一確認した。また、モデル班が作成したモデルについて活性化関数など詳細な設計についても把握するように心掛けた。個人的な反省として、エージェントの開発についてルールを設けなかったことがある。通常 2 人、最大でも 6 人で活動するので通常の会話で十分に同調できるだろうと感じていたが、結果としてプログラムの冗長な部分や統一性のない部分が発生してしまった。これを反省とし、今後の課題としたい。

(※文責: 松田顕)

7.2 小田中嵐

7.2.1 前期

前期の活動を通じてグループリーダーを務めた。慣れない役職ではあったが、メンバーと協力し、当初の計画より大幅に遅れることなく進行することができた。初めは人狼知能を開発するうえで必要な知識の習得、環境構築に努めた。この時点で Java や Git, GitHub に初めて触れ、導入にかなり苦戦した。時間はかかったが、グループメンバーに助けをもらいながら導入することができた。その後、輪読を通じて人狼知能のアルゴリズムや人狼知能に使われる機械学習について学んだ。

中間発表の準備では、プロジェクトの概要を紹介する紹介文を作成した。同時並行で進めたポスター班やウェブサイト班よりも早く完成したため、その後は Google フォームを用いた評価サイトの作成や、評価担当の割り振りをし、中間発表に向けて準備に積極的に活動することができた。中間発表では、質疑応答を担当した。評価には、質問への受け答えがよかったとのコメントもあり、リハーサルで練習した通りしっかりと受け答えができたと思う。

前期の活動を振り返ると、知識の習得がメインになってしまい、実際に手を動かしながら開発まで進めることができなかった。後期に向けて開発を進めていくうえで、さらにプログラミングと機械学習についての理解を深めていく必要がある。

(※文責: 小田中嵐)

7.2.2 後期

後期も引き続きグループリーダーを務め、モデル班としてモデル構築の作業を行った。機械学習のライブラリとして Keras を採用し、Keras についての知識を深めた。ログデータの解析では、占い師および霊媒師の生存人数を発言内容や死亡ログから読み取って返すプログラムを作成した。また、行動決定モデルでは各行動に対する報酬の設定を検討し、一部プログラムを作成した。これらの活動を通して、ニューラルネットワークや強化学習などディープラーニングについての知識を深めることができた。

成果発表会では評価担当の名簿作成、当日のグループ概要の口頭発表およびその資料作成を担当した。念入りにリハーサルを重ね、当日は問題なく発表することができた。

プロジェクト活動を通じて、2つモデルを駆使して戦うエージェントを作成することはできたが、思うような結果が得られず、チームで開発することの難しさを感じた。2つの班で同時に作業を進めることによってすれ違いが起こったこと、メンバー間の能力差によって作業量に差が出てしまったこと、これらに加えスケジュールが遅れ気味であったという全体的な見通しの甘かったことなどが問題点として挙げられる。これらに対し、メンバー間でもっと綿密な情報共有や知識共有を行ったり、うまくタスクを分散させたりするなど、グループリーダーとして臨機応変に対応し活動していくべきであったことが反省点として挙げられる。開発としては技術的に至らない部分が多く、進行手順や計画について開発経験や技術力のある他のメンバーに任せることが多かったが、それぞれのメンバーが開発担当、計画担当、雑務担当のように異なる役割を担当し、役割が個人に集中することを避け、仕事を分散させるという観点からすれば、この方法は有用な進行方法のひとつであったと言える。しかし、それは統率者ありきの話で、リーダーにはすべてを把握し、まとめ上

げるスキルが求められると感じた。今後このようなチームで開発を行うような活動がある際には、この反省を活かしたい。

(※文責: 小田中嵐)

7.3 西村和馬

7.3.1 前期

Git や Python など、開発に必要な環境の構築を行った。前期の活動期間を通し、参考図書 [3] の輪読を行い、エージェント開発の手順や、サポートベクトルマシンの学習を行った。それに並行し、過去の大会に出場したエージェントの役職推定や行動決定に使われるアルゴリズムの調査、開発に利用する機械学習の手法の検討を行った。7月には、人狼知能プロジェクトが主催する人狼知能セミナー 2020 に参加した。そこで模擬大会が行われるということで、それまでに学習したことを活かし、少しでも自分のオリジナル要素が含まれたエージェントを開発しようとしたが、自分の中で納得するものは作れず、結果も振るわなかった。その後、セミナーで学習したことや、模擬大会の結果をもとに、今後開発をするエージェントのアルゴリズムの検討を行った。夏季休暇終了前の段階ではアルゴリズムが決定していないので、しっかりと検討していきたい。中間発表では、ウェブサイトの作成と、後半のスライド発表を担当した。前期を通し、知識や技術の習得が活動の主体であった。自分は意見を述べることや提案をすることが少なく、グループ活動にあまり貢献できていなかったのではないかと痛感している。後期から開発が主体となるので、そこではグループを引っ張っていけるよう、積極的な活動をしたい。

(※文責: 西村和馬)

7.3.2 後期

後期は、モデル班に所属し、主にモデル構築の作業に携わった。ログデータの解析では、肯定発言と否定発言、投票変更数を読み取り、それらの数をリストに格納するプログラムを作成した。行動決定モデルでは、報酬の検討を行った他、騙りの実装など、一部のプログラムの作成を行った。後期の活動を通し、機械学習の制作に関する知識を習得することができた。また、話し合いの際、前期と比べ、発言することが多くなり、以前より積極的に活動することができるようになったと感じた。その一方で、チームでの開発の経験がなかったこと、機械学習を用いたプログラムを制作した経験がなかったことから、話し合いの進行や、軸となる部分の制作を他のメンバーに任せてしまうことが多くなってしまった。また、理解が追いつかず、何度も同じような質問をして話し合いの進行を妨げてしまったことや、Python のプログラムを書くことに慣れておらず、プログラムの制作の際、他のメンバーよりも多くの時間を費やしてしまったことなど、チーム全体に迷惑をかけることが多くなってしまった。

最終成果発表会では、前半パートの質疑応答を担当した。事前準備として想定質問を何個か用意した。しかし、本グループに対しての質問は想定外のものであった。その際、うまく自分ではまとめることができず、他のメンバーに助け舟を出してもらうことになってしまった。今後、臨機応変に対応する力を身に着けることが、必要であると感じた。

7.4 工藤光矢

7.4.1 前期

前期の活動では輪読進行担当を努めた。まず、知識共有のために輪読する参考図書全員で話し合い、『人狼知能で学ぶ AI プログラミング』[3]を参考図書に決定した。その後、輪読の担当を決める際には、活動計画の日程や、参考図書のページのまとまりを考えながら、無理のない範囲で輪読の担当を決め、輪読を進めることで機械学習、人狼知能についての知識を深めることができた。グループ全員で参加した人狼知能セミナー 2020 では、実際のプログラミングを通して、java での人狼知能の開発方法を学んだ。その際、輪読で学んだ知識を活かすことができた。また、セミナーの最後に行われた模擬戦では、奇跡的にはあるが好成績を残すことができた。その他の活動として、中間発表会の準備では、ウェブサイトの作成を行った。GoogleSite という Google の提供するサービスを用いてグループを紹介するウェブページを作成し、デザインの面でこだわることができなかったが、グループの内容、人狼ゲーム、人狼知能プロジェクト [1] についてまとめることができた。

前期は新型コロナウイルスの影響もあり、全面的にオンラインでの活動で作業がスムーズに進まない場面がいくらかあった。前期は知識共有や環境構築などの作業が主であったが、後期は実際に開発となる。プログラミングには自身があるので、開発に積極的に取り組み、遅れを取り戻していきたい。

(※文責: 工藤光矢)

7.4.2 後期

後期ではモデル班を担当し、モデル構築に関する作業を行った。作業内容として、ログデータの抽出では、Python を用いて占い結果、被占い結果などの情報をログデータから抽出するプログラムを作成した。人狼推定モデルの構築では、Keras についての知識を深めながらモデルの構築作業を行い、そのほかに人狼推定モデルの出力部に必要な人狼組み合わせ ID の生成や、完成した人狼推定モデルの学習を実行する作業なども行った。また、実験的に簡易的な役職推定モデルの作成や、メソッドの作成などを行った。ニューラルネットワークや Python を利用するのは初めてだったため、始めのうちはコードの記述や、ニューラルネットワークの学習に苦悩していたが、徐々に活用していくことができるようになった。完成したエージェントの勝率はあまり良い結果を残すことができなかったが、このプロジェクト活動を通して、グループ開発や、機械学習、ニューラルネットワークなどの知識を深めることができたので、有意義な活動であったと感じる。

成果発表会では、Web サイトの作成を準備で行い、当日の成果発表会ではスライドを用いて、発表を行った。当日は問題なく発表することができたが、Web サイトについては見易さが考慮出来ていなかったと考える。図や詳細な情報などを分かりやすくまとめていたつもりであったが、最終発表会の評価において、Web サイトが見づらい、内容をもっとまとめたほうが良いなどの評価を受けた。今後の課題として、Web サイトやその他の掲示物を作成する際には、読み手に考慮しながら作成できるように努力したい。

後期は、全面オンライン活動だった前期とは違い、水曜日はオフライン、金曜日はオンラインの活動だった。前期の時点でスケジュールの過密といった問題が懸念されていたが、全面オンライン活動だった前期と比べ、後期はオフライン活動があるため、スムーズに進みスケジュールの過密が解消されると考えられていた。しかし、スケジュールの過密を解消することはできなかったため、より綿密なスケジュール管理ができればよいと思った。

(※文責: 工藤光矢)

7.5 根上樹

7.5.1 前期

前期活動の全編を通して書記を担当した。記録をとる上で他者の発言理解や技術の内容理解を素早く行う必要があり、深く議論に集中することができた。また可能な限り議論の進行が滞ることを防ぐため、意見や方向性の確認を発するように心がけた。特に現状の問題点や解決策を提示することに重きを置いていた。7月以前に行われた技術調査および習得に関しては、他グループメンバーの手が届きにくそうな部分を中心に行った。また Java や Git の環境構築については事前知識を最低限保持していたため、サポートに回ることができた。Python に関しても夏季休暇を通して、理解を深めることができた。ただ、人狼知能セミナー 2020 に参加した際に、開発スピードの遅さを実感した。後期活動を通して改善していきたいと考えている。他の活動として、中間発表の際には司会進行を行った。タイムキーピングの失敗が痛手であったが、それ以外の進行はスムーズに行うことができた。

現状のプロジェクト進行速度は時勢の問題があれど、やや遅れ気味であると考えており、後期には積極的な開発作業を推し進めたい。特に具体的なモデル選定後の調整作業に重きを置くことを考えている。作業と並行して、有効に利用できる各技術の更なる習得にも努めたい。

(※文責: 根上樹)

7.5.2 後期

後期ではモデル班に割り当てられ、主にモデルのコーディングを担当した。特にプログラムの順序付けや部品分解、モデル本体部である機械学習工程の調整、全体のデバッグを行った。また各種モデルの構築において、理論の調査・プログラムの正当性の検証に力を入れた。メンバーに対するプログラム内容の解説・共有や、設計班との連携活動も行った。

5人以上でのチーム開発の経験をそれほど持っていなかったこともあり、他メンバーの記述したコードとの重複や、依存性の解決に比較的大きい割合の時間を費やしてしまった。また読みやすいコーディングを心掛ける必要性に関しても、身をもって体感することとなった。行動決定モデルの作成において、有意な結果を導き出せなかったことは心残りであるが(特に力を入れて製作していた機能である分余計に)、人狼推定モデルや行動決定モデルの作成を通して、機械学習を利用・開発するための技術を習得することができ、有意義な経験として蓄積されたと感じる。

成果発表会においてはモデルに関する質疑応答を担当した。事前準備については比較的十分にできていたと考えるが、想定を外れた質問に対する臨機応変さに欠けていたと感じる。自分の中の知識・理解を安定して引き出すための冷静さの獲得や、事前想定の変更の練りこみが課題である。

7.6 木村高之

7.6.1 前期

5, 6月はプロジェクトを行うための環境構築 (Git, GitHub, Java, Python), 参考図書とし『人狼知能で学ぶ AI ディープラーニング』[3] の輪読・知識共有を行った。環境構築の際, 私は Git や GitHub について初めて知ったので, わからない部分はグループメンバーに聞いて作業を行った。7月には人狼知能セミナーにグループ全員で参加し, Java での人狼知能の作成方法と Python での人狼知能の作成方法を学んだ。その後, 本グループでは, Python でプログラムを書いていくことが決まった。そのとき, 私は Python についての知識がなかったので, プライベートで Python の本を一冊買って勉強をすることにした。今まで大学で学習してきたプログラミング言語と共通点多かったので, 一週間ほどで Python の内容をだいたい理解することが出来た。また7月には, 参考図書の輪読継続・過去の人狼知能大会のプログラムを調べた。前期はコロナの影響もあり, すべての活動がオンラインで行われ, 作業がスムーズに進まないことも多いように感じた。後期はすべてがオンラインというわけではなく, 集まる機会が多少はあるとのことで, その機会を活かし, 本グループの最終的な目標である大会でいい成績を残せるように努めていきたいと思う。

(※文責: 木村高之)

7.6.2 後期

後期では, 前期で学習したことを元にモデル班と設計班に分かれて, 人狼知能エージェントを実装する作業に取り掛かった。その際, 私は設計班に所属し, モデル班が作成した機械学習のモデルを人狼エージェントに適応させる作業を行った。この作業の具体的な内容としては, 人狼ゲームを行うサーバーから送られてくる発言・行動などのログデータの収集, 収集したログデータから生存人数や関係性などの解析, 解析結果を特徴量として人狼推定モデルが読み込める形式への変換, 行動決定モデルに渡す特徴量の収集・解析・変換, 行動決定モデルの出力から実際に行動を行う処理であった。これらの作業を行う前に, pandas や Keras についての知識がなかったので, pandas と Keras の勉強を行った。

後期の成果発表では, 司会を担当した。前期でも司会をやっていたこともあり, だいたい勝手が分かっていたので問題なく進行出来ていたと思う。

また, 私はプログラミングがあまり得意な方ではなかったので, プログラムを書くのに時間がかかり, 活動時間内に終わらないことも多々あった。それを補うために土日に作業を行っていた。今回の設計班での作業を通して, どのような過程でプログラムを書いて行けばよいのかを理解することができ, 今回の経験は今後生きていくものだと感じた。

(※文責: 木村高之)

第 8 章 まとめ

8.1 プロジェクト活動の成果

前期のプロジェクト活動では、人狼知能エージェントの構成の仕方や機械学習全般の基礎知識を参考図書（『人狼知能で学ぶ AI プログラミング』[3]）の輪読での知識共有、人狼知能を作成するうえで必要となる環境構築（Java 言語、Python 言語、Git、GitHub の導入）を行った。また、人狼知能プロジェクト [1] 主催の人狼知能セミナー 2020 に参加し、基本的な人狼知能エージェントの作成方法、Java と Python での開発方法を学んだ。そこで、現在は役職推定に機械学習を搭載した、行動決定をルールベースで行うエージェントが主流であることを知った。今後の方針として、エージェント作成に使用する言語として Python 言語を用いること、エージェントの行動決定に機械学習の判断を導入すること及び機械学習モデルにはニューラルネットワークを用いることを決定した。中間発表では反省点として、Zoom での画面共有がうまくいかなかった点、動画がないため具体的なイメージがしにくかった点、人狼知能大会の説明が詳しくされていなかった点、活動内容がわかりにくかった点など後期発表で活かせる多数のフィードバックをもらった。

後期のプロジェクト活動では、モデル班と設計班に分かれ、人狼知能エージェントの実装に取り掛かった。モデル班は、ニューラルネットワークを用いた人狼推定モデルや DQN(Deep Q-Network) を用いた行動決定モデルの実装を行った。その具体的な活動としてはログから学習データの抽出、人狼推定モデルの実装、行動決定モデルの実装である。ログから学習データの抽出は、過去に開催された人狼知能大会のログデータから、学習に必要な特徴量として、日付、生存者数、占い師数、霊媒師数、占い師 CO 状況、霊媒師 CO 状況、占い結果、肯定発言数、否定発言数、投票変更数の抽出を行った。人狼推定モデルは Python の機械学習ライブラリである Keras を用いてニューラルネットワークのモデルを構築し、その完成したニューラルネットワークに、ログから抽出した学習データと正解データ（誰が人狼であるか）を利用して学習を行った。その結果として、最大 15 人の人狼ゲームにおいて、人狼推定の精度が 63 %程度のもので完成した。行動決定モデルは、人狼推定モデルやサーバーから送られてくるゲームの情報から DQN(Deep Q-Network) を用いて最適な行動を決定するエージェントを作成したが、そのエージェントは勝率 43.3 %とランダムに行動を選択したものと大差ない結果になった。その原因として、特徴量の不足、学習回数の不足、行動とそれに対する報酬の多様化が挙げられ、その改善案として、モデルの設計の見直し、毎回同じ対象に同じ行動をしていることに対して負の報酬を与えて改善の試み、学習回数の増加が必要であると考察した。また、設計班は事前準備として、pandas, Keras などの Python と機械学習についての調査と学習を行い、その後、人狼エージェントに機械学習を適用させる作業を行った。その具体的な作業として、人狼ゲームを行うサーバーから送られてくる発言・行動などのログデータの収集、収集したログデータから生存人数や関係性などの解析、解析結果を特徴量として人狼推定モデルが読み込める形式への変換、行動決定モデルに渡す特徴量の収集・解析・変換、行動決定モデルの出力から実際に行動をとる処理を行った。

(※文責: 木村高之)

8.2 改善点

9月現在での問題点は、Pythonでのエージェント作成の作業がほぼ全く進行していないことである。後期はこれが主な活動内容となるが、残っている活動期間はあまり長くないこと、細かな調整を行う必要があることを考えると、急速にプロトタイプ作成の作業を進める必要がある。スケジュールの工夫・改善や、疑問点の共有や前提のすり合わせ、徹底した役割分担を行うことにより、スムーズに作業を進めることが望まれる。時間に余裕ができ次第、エージェントの動作を視覚的に周知させることのできる仕組みや図解の作成を行うことも望ましい。また、機械学習モデルには、ニューラルネットワークが用いられることが決定しているが、当初の形式に囚われず、臨機応変にモデルを選択する予定である。

後期にはエージェントが完成した。人狼の推定については、細かい調整や、入力の特徴量を増やすことでさらに精度の向上が見込まれる。行動決定については、ランダムに行動を選択したものと勝率が差なかった。原因として、特徴量の不足、学習回数の不足、行動とそれに対する報酬の多様化が挙げられた。それをもとに、モデルの設計の見直し、毎回同じ対象に同じ行動をしていることに対して負の報酬を与える、学習回数を増やすなどの改善が必要であると考察した。

後期のグループ活動における問題点として、開発スケジュールの過密が挙げられる。前期に開発環境の構築に想定以上の時間を費やしてしまったことが主な原因であると考えられる。前期は全面オンラインでの活動であり、スムーズに作業を進めるのは比較的困難な状況であったとはいえ、逐一各メンバーの進行状況を確認するなど開発環境の構築にかかる時間を減らし、開発作業にある程度ゆとりを持たせることが可能だったと考える。

また、スケジュールの過密によって引き起こされた問題も多発した。エージェントの開発を行う際、ルールを設けずに開発を進めてしまった結果、プログラムの統一性がない部分が発生した。それにより、モデル班と設計班の間で情報共有をする際に、お互いの制作しているプログラムを理解するのに時間がかかってしまった。また、統合の際、少し手間取ることになってしまった。これらの班分けによる問題点は、時間をかけてグループ全体で話し合い、ルールを設けることなどによって、解決することが可能である。また、複数のエージェントを制作して比較など、当初予定していた作業が白紙化されたことや、2つのモデルの性能アップに必要な時間をとることができなかった、といった事態も発生した。これらもスケジュールの過密によって起こった問題であるといえる。やはり、前期の段階で、時間にゆとりをもって開発に取り組む見通しを立てていれば、当初の予定通り開発を行うことができたと考えられる。

(※文責: 西村和馬)

参考文献

- [1] 人狼知能プロジェクト公式サイト. <http://aiwolf.org/>. (Accessed on 17/09/2020).
- [2] 情報処理学会. コンピュータ将棋プロジェクトの終了宣言. <http://www.ipsj.or.jp/50anv/shogi/20151011.html>. (Accessed on 17/09/2020).
- [3] 狩野芳伸, 大槻恭士, 園田亜斗夢, 中田洋平, 箕輪峻, 鳥海不二夫 (著), 人狼知能プロジェクト (監修). 人狼知能で学ぶ AI プログラミング. マイナビ出版, 2017.
- [4] 稲葉通将, 片上大輔, 狩野芳伸, 大槻恭志. 人狼知能と不完全情報ゲーム. 人工知能, Vol. 34, No. 6, pp. 876–880, nov 2019.
- [5] Keras documentation. <https://keras.io/ja/>. (Accessed on 10/2020).
- [6] Chainer tutorial 11. pandas 入門. https://tutorials.chainer.org/ja/11_Introduction_to_Pandas.html. (Accessed on 10/2020).
- [7] Aiwolfpy v0.4.9. <https://www.slideshare.net/HaradaKei/aiwolfpy-v049>. (Accessed on 10/2020 - 12/2020).
- [8] Aiwolfpy. <https://github.com/aiwolf/AIWolfPy>. (Accessed on 10/2020).
- [9] 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. 2F41–2F41, 2016.
- [10] 大川貴聖, 吉仲亮, 篠原歩. 深層学習を用いて役職推定を行う人狼知能エージェントの開発. ゲームプログラミングワークショップ 2017 論文集, No. 2017, pp. 50–55, nov 2017.
- [11] DQN の理論説明. https://www.renom.jp/ja/notebooks/tutorial/reinforcement_learning/DQN-theory/notebook.html. (Accessed on 10/2021).