The 33rd Annual Conference of the Japanese Society for Artificial Intelligence, 2019

Using Q-learning and Estimation of Role in Werewolf Game

Makoto Hagiwara*1 Ahmed Moustafa*1 Takayuki Ito*1

*1Nagoya Institute of Technology

This paper introduces a novel construction strategy in Werewolf Game using reinforcement learning(RL). Werewolf Game is a type of incomplete information games in which the final results of the game is linked to the success or failure in communication. In this paper, we propose a model that uses RL and estimating other agent's role in order to learn playing strategy in Werewolf Game. In the proposed model, RL is used for deciding the actions of the learning agent and Naive Bayes classifier is used in order to estimate other agent's role. Up till now, there is no previous research that has effectively applied RL in Werewolf Game among existing AIwolfs in large scale environments. Therefore, by combining RL and estimation of other agent's role, we demonstrate through experimentation that the proposed approach achieved high level of performance in 11 people Werewolf Game.

1. Introduction

In this paper, we investigate the playing strategy of Werewolf Game, a cornerstone of communication games. Werewolf Game has two competing teams-villagers and wolves. In Werewolf Game, execution by vote or attack by werewolf eliminates the player from the game. The villagers win if all the werewolf players are eliminated. The werewolves win if enough number of villagers are eliminated so the number of villagers is even to the number of werewolves. In this regard, conversation becomes important to distinguish wolves from existing players.

In Werewolf Game, natural language is used for conversation. However, the processing of natural language by artificial intelligence (AI) remains a technical challenge. Towards this end, Osawa *et al* [Oosawa 14] have developed a special protocol for enabling AIwolf to conduct conversation. In this context, Toriumi *et al* [Toriumi 14] released a construction kit for developing intelligent agents that play the Werewolf Game using its developed protocol. In specific, Toriumi *et al* [Toriumi 14] attempted to solve the challenges for realizing sophisticated communication by developing an environment in which the intelligent agents play the Werewolf Game and gain collective intelligence via a contest. In this regard, a contest that involves the Werewolf Game was recently held for the first time as part of the Computer Entertainment Developers Conference (CEDEC2015) in Japan, with over 50 teams participating. Therefore, this paper adopts the conversation protocol proposed in [Toriumi 14] and the construction kit proposed in [Oosawa 14].

In the AIwolf contest held in CEDEC2018, the win rate of werewolf in the contest was about 30%. In this regard, Inaba *et al* [Inaba 12] researched the win rate in Werewolf BBS and estimated the percentage of this win rate. BBS is a web application for playing Werewolf Game on the Internet. The Construction kit that is proposed in [Oosawa 14] is built based on BBS. In specific, Inaba *et al* [Inaba 12] suggested that win rate of werewolf is about 50% in the same distribution of roles as that of the contest. Toriumi *et al* [Toriumi 16] suggested that AIwolf has shown performance to learn the win strategy in particular to Werewolf Game, and suggested that AIwolf has strong potential to play Werewolf Game. On the other hand, the win rate of werewolves in the contest held in CEDEC2018 demonstrated that the performance of werewolves in AIwolf is worse than that of villagers in AIwolf. In additon to this, only when the agents are assigned to the role of werewolves, these agents could learn game strategy not as single agents but as a group of agents. This is because information about the member or number of werewolves is known to the players of werewolves in the begining of the game. This factor is very important in Werewolf Game. Therefore, this paper focuses on improving the performance of werewolves in AIwolf.

It is difficult to use common game AI methods, such as tree search algorithms, because Werewolf Game is played through conversation with werewolf agents, i.e., AIwolf. Therfore, in Werewolf Game, it becomes necessary for AIwolf to adopt a voting strategy, because recent research [Doguro 18] has shown that voting information is linked to the estimation of werewolves. In this regard, Doguro and Matsubara [Doguro 18] suggested evaluating the effectiveness of voting information in estimating werewolves. As a result, it becomes important to define such actions as voting depends on game condition. Most of the existing AIwolf agents decide their actions by rule-based programs. Since WereWolf Game has variable conditions which depend on votes or the number of existing players, large state space is necessary to describe them. As a result, it becomes difficult to describe all proper actions in the game by rule-based programs. Therefore, it is important for AIwolf to learn the strategy in Werewolf Game by themselves.

In order to overcome this problem, we employ RL [Sutton 98], because RL is suitable for choosing a sequence of actions that lead to a strategy. In this regard, RL enables AIwolf to build a game playing strategy based on the sequence of these actions that exist in each state of Werewolf Game, and we propose that RL promotes the performance of AIwolf to succeed in the construction of game playing strategy.

In Werewolf Game, it is important to estimate the roles of other players, because information about the roles of other

Contact: Makoto Hagiwara, Nagoya Institute of Technology, hagiwara.makoto@itolab.nitech.ac.jp

players is important in the voting process. For example, estimation of role gives villagers information that tells them who are werewolves. This information enables the villagers to vote in order to decide the werewolf. In this regard, Toriumi *et al* [Toriumi 16] suggested correlation between voting and win rate in Werewolf Game played by AIwolf. Therefore, we expected that adopting estimation of other agent's role develop AIwolf. In estimation of other agent's role, Kaziwara *et al* [Kaziwara 16] suggested the effectiveness of SVM and Ookawa *et al* [Ookawa 17] suggested the effectiveness of deep learning. In the AIwolf contests held in CEDEC2017 and CEDEC2018, the AIwolf with Naive Bayes classifier won and adopted Naive Bayes classifier to estimate other agent's role.

In this paper, we employ these two methods, RL and estimation of other agent's role, to construct werewolf's strategy of action. Since there is no previous research to suggest the effectiveness of RL in Werewolf Game in large scale with existing AIwolf. Therefore, we examined the effectiveness of RL in Werewolf Game in 11 player with existing AIwolf.

2. Preliminaries

2.1 Werewolf Game

Werewolf Game is a conversation-based party game that is played using the communication abilities we possess as humans. When the game starts, all the players are randomly divided into either the villager side or the werewolf side, and then allocated to different roles. We summarize these roles in Table 1. The players in the villager side and possessed players do not know which side other players belong to; however, the werewolf players know the players of wolves.

In Werewolf Game, there are two phases, i.e., day phase and night phase. In the day phase, all players discuss who the werewolf are, and then select a player who becomes target for execution based on voting. In the night phase, the werewolf players attack a human player that is selected based on discussion with other players of the werewolf. Other players with specific abilities of their role can use their abilities at night. The executed player and the attacked player are eliminated form the game, and are not allowed to participate in further discussions or votes. Those player roles are not revealed until the game is over. A crucial aspect for villager players is to detect the lies presented by werewolf players in discussions. A crucial aspect for werewolf players is to manipulate discussion to their advantage by impersonating a role.

The villagers win if all werewolf players are eliminated. The werewolves win if enough number of villager players are eliminated so the number of villager players are even to the number of werewolf players.

2.2 Q-Learning

Q-learning [Watkins 92] is a model-free RL algorithm. In this context, Q-learning provides the learning agents with the capability of learning to act optimally in markovian environments by experiencing the consequences of their actions, without requiring these agents to build explicit models of their environments. In Q-learning, Q values estimate the values of actions and decide the learning agent actions. Through experiencing the consequences of their actions, Q-learning agents change Q values. In so doing, the Q values are updated using Equation (1):

role	ability
Seer	Every night, selecting one active player, you can know whether the player is werewolf or not.(divination)
Medium	Every night, you can know whether the player executed is werewolf or not.(medium result)
Knight	Every night, selecting one active player, you can guard the player.(guard) If target for guard is same as target for attack, player attacked is not eliminated from the game.
Werewolf	Every night, selecting one active player, you can attack the player.(attack)
Villager	the player of villager side. They don't have ability.
Possessed	the player of werewolf side. They don't have ability.

Table 1: Role ability

$$Q(s_t, a_t) = r_t + \gamma \max Q(s_{t+1}, a) \quad (1)$$

In this equation, s_t is the current state. a_t is the current action. γ is the learning rate. s_{t+1} is the next state. In Q values of next state, the highest value is used for updating Q values of the current state.

3. Proposed Model

First, we define state, reward and action in the proposed model as follows.

Definition One: Action

In the proposed model, three types of actions are available which are vote, attack and pronunciation. Vote is for choosing one player to eliminate from the game. Attack is for choosing one player to attempt to kill. Pronunciation is for choosing one pronunciation from the pronunciations set that is defined by the game protocol. This set of pronunciations also contains SKIP. SKIP means that I have nothing to talk. In the proposed model, the target of action consits of the player whose probability of being in some role is the highest among all players, the player who gets the most declarations of vote, the player who is not divined, the player whose results of divination include a result of being werewolf, the player whose results of divination include only result of being villager, the player whose time of votes of being werewolf is the highest among all existing players, and finally, any random player.

Definition Two: Pronunciation

In the proposed model, the pronunciation set that is defined in the game is limited by the game protocol. As shown in Table 2, the number and type of available pronunciations is limited. There are other types of pronunciations, however most of the existing AIwolf agents don't consider those types of pronunciations. Therefore, those pronunciations are not considered in the proposed model.

Definition Three: State

In the proposed model, there are six types of states which

type	example
Estimation	I think that PlayerA is werewolf.
vote declaration	I'll vote to PlayerA (this may be fake declaration.)
Result of ability	I divined PlayerA and found that PlayerA is werewolf.
Coming out	My role is Seer.

Table 2: Explaination about pronunciation

are day, turn, the number of players who declared that their role is seer (or medium) in discussion, the number of existing players of villager and werewolf, whether each werewolf is divined (boolean) and present number of declarations of vote to each werewolf.

Definition Four: Reward

The proposed model receives reward in the following three cases which are: when game is over, when attack is failed, when the player with ability or the player of werewolf side is eliminated.

The proposed model adopts Naive Bayes classifier to estimate other agents's roles based on the frequency of actions of every turn in the past games. As shown in Figure 1, we implemented the Naive Bayes classifier algorithm that is used by Cndl [Agent], which is an existing AIwolf agent, as a reference model. In addition, we present an algorithm that demonstrates the learning model in Algorithm 1.

Algorithm 1 Learning Model

```
function GETACTION(state)
    if Math.random() < ϵ then
        return action randomly
    else
        return action selected from top-3 Q value actions
function UPDATEQLIST
    for (s_t, s_{t+1}, a_t, r_t) in latest episode do
        Q(s_t, a_t) = r_t + γ max Q(s_{t+1}, a)
```

As shown in Algorithm 1, the proposed model calls *GETACTION* function, when the proposed model selects its action. In *GETACTION* function, there is an ϵ percent chance that the next action is selected randomly. Other than that, the proposed model selects the next action from the three available actions with the highest $Q(s_t, a_t)$ values in a_t using Boltzmann selection. When the game is over, the proposed model calls *UPDATEQLIST* function to update the Q values based on the sequences of (s_t, s_{t+1}, a_t, r_t) in the latest episode.

4. Experiment

4.1 Setup

We performed a set of experiments involving the rule-based model and the proposed RL model to show the efficiency of the proposed model. In these experiments, we used the AIwolf Server (ver0.4.11) [Oosawa 14]. In addition, the opponent agents (contest agents) that are used in the AIwolf competitions are built using Java Technology. In these experiments, we compared the win rate of the proposed RL model with that of the rule-based models, i.e., Cndl and Udon [Agent], when each player whose role was

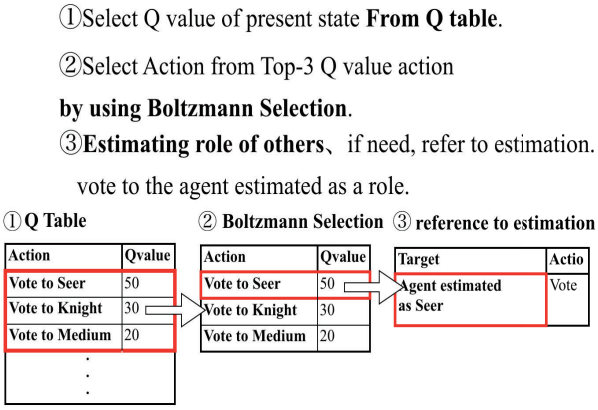

Figure 1: Overview of action decision in proposed model

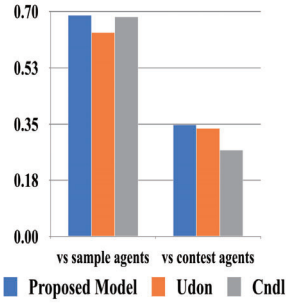

Figure 2: Result

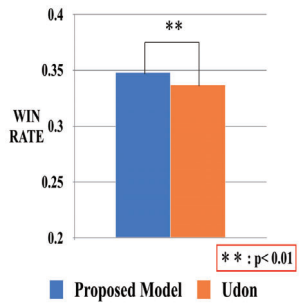

Figure 3: T-Test

werewolf played the game. It is important to note that Cndl won the 2017 AIwolf contest. In addition, Udon showed the highest win rate of werewolf in 2017 AIwolf contest. We executed 100,000 games among 10 sample agents (In this context, sample agents act almost randomly.) and an evaluation agent and executed 100,000 games among 10 contest agents and an evaluation agent. In this regard, Evaluation agent is the proposed RL agent or cndl or Udon. As for pre-training, we executed 2,000,000 games with sample agents and 50,000 games with contest agents.

4.2 Results

We demonstrate the results of these experiments in Fig 2. As shown in Figure 2, Udon's win rate against sample agents is lower than the win rate of both cndl and the proposed RL agent. In addition, we have conducted t-test and present the result of t-test in Fig 3. As shown in Fig 3, significant difference is shown between the proposed RL model win rate and Udon's win rate. In addition, we employed DQN instead of Q-learning in the same learning way as Q-learning. However, the win rate in the games among sample agents is higher but the win rate in the games among contest agents is lower than that of agent using Q-learning.

4.3 Discussion

In the game with sample agents, Udon has lower win rate than that of cndl [Agent]. However, in the game with contest agents, Udon has higher win rate than that of cndl. This is because Udon [Agent] adopts strategy to take in action of villager. Because sample agents decide target for vote and attack randomly, this Udon's strategy diminish their win rate versus sample agents. However, Udon's win

rate versus contest agents are better than cndl's win rate. These win rate suggest the effectiveness of Udon's strategy in Game with contest agents. Contest agents are different from sample agent in that their strategy is sophisticated enough that they adopt such strategies as power play (To decide target for execution by organized vote of werewolf side) or roler (To execute all agents who declared CO of specular role). The proposed RL agent showed better win rate than Agent cndl and Agent Udon. It is assumed that the pre-train in the games with sample agents promote the performance of the proposed model in the game with existing agents. In pre-train, the proposed model learned how to select the target for vote or attack in the game with sample agents. This helps the proposed model to select the target for vote or attack in the game with contest agents. As shown in Fig 3, there is a significant difference($p<0.01$) between the proposed model and Udon. Therefore, the proposed model showed that the performance of RL is no less than that of rule-based program in the Werewolf Game with existing model. In addition, we need adjustment to use DQN instead of Q-learning.

5. Related Work

Kaziwara *et al* [Kaziwara 14], [Kaziwara 15]; and Wang and Kaneko [Wang 17] researched Q-learning in AIwolf. In specific, Kaziwara *et al* [Kaziwara 14] adopted another protocol that is different from the proposed protocol by the authors in [Toriumi 14] and suggested the effectiveness of RL in Werewolf Game. In Kaziwara *et al* [Kaziwara 14], their agent used RL to learn the object to attack, vote and use ability of role. Their agent also used RL to learn choosing the role for impersonating their role when their roles are werewolf or possessed.

In Kaziwara *et al* [Kaziwara 15], the effectiveness of RL was demonstrated in the environment that adopts the proposed protocol in [Toriumi 14], and that selects opponent as sample agents. In this context, sample agents act almost randomly. In their work, they have defined state as "the probable collection of role based on talk in game" and defined action as "the target for attack, vote and use of ability", "condition of telling CO" and "impersonating a role in werewolf or possessed". In Kaziwara *et al* [Kaziwara 15], action is decided by multiplication of value of Q by the frequency of state. In their work, Kaziwara *et al* [Kaziwara 15] consider some types of roles but they did not consider all types of roles in Werewolf Game. Therefore, considerating all kinds of role in Werewolf Game, the proposed approach adopts a novel strategy that defines state using estimation of other agent's role and action using Q-learning.

On the other hand, Wang and Kaneko [Wang 17] defined action as "the target to which attack, vote and use ability of role", and defined state as CO, impression to other player(deceive, reliance and nothing) and role cleared in game. Both Kaziwara *et al* [Kaziwara 14], [Kaziwara 15] and Wang and Kaneko [Wang 17] employed heuristic method in state and action. Conversely, Wang and Kaneko [Wang 18] did not employ heuristic in action. In specific, Wang and Kaneko [Wang 18] employed DQN. No other research employed DQN in AIwolf. In this regard, Wang and Kaneko [Wang 18] suggested the effectiveness of DQN without heuristic action in Werewolf Game among 5 players with existing AIwolf. However, the proposed approach considers Werewolf Game among 11 players is more complex than that among 5 players.

6. Conclusion

This paper proposes a novel model that adopts RL for constructing a winning strategy and Naive Bayes classifier for estimating other agent's role in Werewolf Game. In the proposed model, the learning agent decides its actions based on RL. In addition, the learning agent decides the target for these actions based on estimation using Naive Bayes classifier. Thorough a set of evaluation experiments, the proposed model showed higher win rate than other existing models especially in 11 player Werewolf Game. The future work is set to investigated the necessary approaches that are needed in order to decrease the learning time.

References

[Agent] http://aiwolf.org/resource

[Doguro 18] Doguro,H., Matsubara,H.: Effectiveness Evaluation of Vote Information in Estimation Werewolf Using Neural Network, GAT2018, p1-4, 2018.

[Inaba 12] Inaba,M., Toriumi,F., Takahashi,K., et al.: The Statistical Analysis of Werewolf Game Data, GPW2012, p144-147, 2012.

[Kaziwara 14] Kaziwara,K., Toriumi,F., Oohashi,H., et al.: 強化学習を用いた人狼における最適戦略の抽出 (no English title), IPSJ vol2014, No.1, p597-598,2014.

[Kaziwara 15] Kaziwara,K., Toriumi,F., Inaba,M.: Design of Agent in "Are you a Werewolf?" using Reinforcement Learning, The 29th JSAI2015, p1F2-2, 2015.

[Kaziwara 16] Kaziwara,K., Toriumi,F., Inaba,M., et al: Development of AI Wolf Agent using SVM to Detect Werewolves, The 30th JSAI2016, p2F41,2016.

[Ookawa 17] Ookawa,T.,Yoshinaka,R.,Shinohara,A.: Development of AI Wolf Agent Deducing Player's Role Using Deep Learning,GPW2017,p50-55,2017.

[Oosawa 14] Osawa,H.,Toriumi,F.,Katagami,D.,et al.:Designing Protocol of Werewolf Game:Protocol for Inference and Persuasion, FAN2014, p78-81, 2014.

[Sutton 98] Sutton,R.S., Barto,A.G.: Introduction to reinforcement learning(Vol.135), MIT press,1998.

[Toriumi 14] Toriumi.F.,Kaziwara,K.,Osawa, H.,et al.: Development of AI Wolf Server,GPW2014,p127-132,2014.

[Toriumi 16] Toriumi,F., Shinoda,K., Inaba,M., et al.: Analysis of Agent Behaviors in First AI Wolf Contest, IPSJ vol2016-EC-41 No.3, p1-8,2016.

[Wang 17] Wang,T., Kaneko,T.:Comparation of Methods for Choosing Actions in Werewolf Game Agents, GPW2017, p177-182, 2017.

[Wang 18] Wang,T.,Kaneko,T.:Application of Deep Q Network in Werewolf Game Agents,GPW2018,p16-22,2018.

[Watkins 92] WATKINS, C., DAYAN, P.: Q-learning, Machine learning, 1992, 8.3-4: 279-292.