Overview of AIWolfDial 2019 Shared Task: Contest of Automatic Dialog Agents to Play the Werewolf Game through Conversations

Yoshinobu Kano[1,*] Claus Aranha[2] Michimasa Inaba[3] Fujio Toriumi[4] Hirotaka Osawa[5] Daisuke Katagami[6] Takashi Otsuki[7] Issei Tsunoda[1] Shoji Nagayama[8] Dolça Tellols[9] Yu Sugawara[10] Yohei Nakata[11]

[1]kano@inf.shizuoka.ac.jp, itsunoda@kanolab.net, Shizuoka University, Johoku 3-5-1, Naka-ku, Hamamatsu, Shizuoka 432-8011 Japan

2caranha@cs.tsukuba.ac.jp, University of Tsukuba, Tennoudai 1-1-1, Tsukuba-shi, Ibaraki 305-8577 Japan

[3]m-inaba@uec.ac.jp, the University of Electro-Communications, 1-5-1 Chofugaoka, Chofu, Tokyo, Japan 182-0021 Japan

[4]tori@sys.t.u-tokyo.ac.jp, the University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656 Japan

[5]osawa@iit.tsukuba.ac.jp University of Tsukuba, Tennoudai 1-1-1, Tsukuba-shi, Ibaraki 305-8577 Japan

[6]katagami@t-kougei.ac.jp, Tokyo Polytechnic University, 1583 Iiyama, Atsugi, Kanagawa 243-0297 Japan

[7]otsuki@yz.yamagata-u.ac.jp, Yamagata University, Jonan 4-3-16, Yonezawa, Yamagata 992-8510 Japan

[8]nagayama@forest.eis.ynu.ac.jp, Yokohama National University, Tokiwadai 79-1, Hodogaya-ku, Yokohama, Kanagawa 240-8501 Japan

[9]tellols.d.aa@m.titech.ac.jp, Tokyo Institute of Technology, Ookayama 2-12-1, Meguro-ku, Tokyo 152-8550 Japan

[10]suga@ist.hokudai.ac.jp, Hokkaido University, Nishi 5, Kita 8, Kita-ku, Sapporo, Hokkaido 060-0808 Japan

[11]nakata.ud@gmail.com

Abstract

The AIWolf project has been holding contests to play the Werewolf game ("Mafia") by automatic agents for a couple of years. A difficulty of the Werewolf game is that the game is an imperfect information game, where a player's role is hidden from other players. Players are required to infer the roles of other players through free conversations; players of a specific role should tell a lie, while others try to break through lies. We employ this werewolf game as a novel way of evaluations for dialog systems. Because the werewolf game forces players to deceive, persuade, and detect lies, neither inconsistent nor vague response are evaluated as "unnatural", losing in the game. Our werewolf game competition and evaluation could be a new interesting evaluation criteria for dialog systems, but also for imperfect information game theories. In addition, the werewolf game allows any conversation, so the game includes both task-oriented and non-task-oriented conversations. This aspect would provide a handy intermediate goal rather than to create a general dialog system from scratch. In this AIWolfDial 2019 shared task, five participant agents played games in English and Japanese. We performed subjective evaluations on these game logs.

1 Introduction

The AlphaGO [1] system defeated the human champion player in Go. However, AI game player is still far from being successful in the Werewolf game that requires complex communications, in addition to the nature of an imperfect information game, while Go is a perfect information game. Playing the Werewolf game would be the next grand research challenge for the AI players.

"Are You a Werewolf?", or "Mafia" (hereafter "werewolf game"), is a communication game conducted solely through discussion. Players must exert their cognitive faculties fully in order to win. In the game, players must hide information, in contrast to perfect information games such as chess or Reversi. Each player acquires secret information from other players' conversations and behavior and acts by hiding information to accomplish their objectives. Players are required persuasion for earning confidence, and speculation for detecting fabrications.

We employ this werewolf game as a novel way of evaluations for dialog systems. While studies of dialog systems are very hot topics recently, they are still insufficient to make natural conversations with consistent context, or with complex sentences. One of the fundamental issues is a lack of an appropriate evaluation.

Because the werewolf game forces players to deceive, persuade, and detect lies, neither inconsistent nor vague response are evaluated as "unnatural", losing in the game. Our werewolf game competition and evaluation could be a new interesting evaluation criteria for dialog systems, but also for imperfect information game theories. In addition, the werewolf game allows any conversation, so the game includes both task-oriented and non-task-oriented conversations. This aspect would provide a handy intermediate goal rather than to create a general dialog system from scratch.

In order to promote such a research challenge, the AIWolf project [2] has been holding competitions every year to play the Werewolf game automatically. We describe our Werewolf player agent system which participated the AIWolfDial 2019 shared task (the natural language division of the 2019 competition of AIWolf) [3]. The shared task was performed in Japanese and English languages. We automatically translate the system I/O to connect Japanese agents with English agents.

1. Werewolf Game in Shared Task

We briefly explain the rules of the werewolf game in this section. Before starting a game, each player is assigned a hidden role from the game master (a server system in case of the AIWolf competition). The most common roles are "villager" and "werewolf". Each role (and a player of that role) belongs either to a villager team or a werewolf team. The goal of a player is for any of a team members to survive, not necessarily the player him/herself.

While there are many variation of the Werewolf game exists, we only explain the AIWolfDial 2019 shared task setting in this paper.

There are other roles than the villager and the werewolf: a seer and a possessed. A seer belongs to the villager team, who has a special talent to "divine" a specified player to know whether the player is a human or a werewolf; the divine result is notified the seer only. A possessed belongs to the villager team but his/her goal is win the werewolf team.

A game consist of "days", and a "day" consists of "daytime" and "night". During the daytime phase, each player talks freely. At the end of the daytime, a player will be executed by votes of all of the remained players. In the night phase, special role players use their abilities: a werewolf can attack and kill a player, and a seer can divine a player. The victory condition of the villager team is to execute all werewolves, and the victory condition of the werewolf team is to make the number of villager team less than the number of werewolf team. A game in the AIWolfDial 2019 shared task have five players: a seer, a werewolf, a possessed, and two villagers.

In the shared task, Day 0 does not start games but conversations e.g. greetings. A daytime consists of several turns; a turn is a synchronized talks of agent, i.e. the agents cannot refer to other agents' talks of the same turn.

An AIWolf agent communicates with an AIWolf server to perform a game. Other than vote, divine, and attack actions, an agent communicates in natural language only. An agent may insert an anchor symbol (e.g. ">>Agent[01]") at the beginning of its talk, in order to specify which agent to speak to.

2 Participant Systems

Five participants provided AIWolf agent systems. There were five Japanese systems and one English system. As we performed five players games, inputs and outputs of Japanese systems were translated into/from English by the Google translate service to play with the native English agent. We briefly describe designs of each system below.

2.1 CanisLupus

Team *CanisLupus* created an agent that talks like a detective in a mystery novel. This agent determines its behavior based on the standard tactics of the werewolf game and its preferences toward each agent. This agent consists of the following modules: an interpretation module that determines the meaning of a statement and translates it into intention like protocol branch, a generation module that translates intention into natural Japanese language, an affection module that records preferences for each agent, and a central module to coordinate these interpretation module.

Using MeCab, their system morphologically analyze the words and determine the meaning of the sentence. For example, if all the words "divined", "Agent [xx]", "werewolf" are included, they can infer that the sentence means "DIVINED Agent[xx] WEREWOLF".

The generation module receives the type of speech from the central module and converts it into the natural Japanese language using a large number of prepared template sentences. For example, if you call this module like "generate ("declare _ VOTE", 1)", their utterance template for "declaration of voting" will be randomly selected. It then performs a substitution on the agent name given to the argument and finally returns the statement "I' m going to vote for Agent [01] tonight."

The affection module records the preferences to each agent. 18 pairs of reason and weight like "You voted for the people who I loved: - 4 points" are set in advance, and the number of times is accumulated as the corresponding situation occurs. When this agent decides whom to vote for, who has the lowest total of the product of the number of occurrences and the weight of each reason is selected.

The central module coordinates the other modules described above. The agent makes most decisions based on the standard tactics implemented in this module.

2.2 Dreaming

Team *Dreaming* created implemented their agent in Java. There are two versions of the agent so that it can play against agents communicating in English or Japanese. Both versions follow the same game strategy but have conversational capabilities adapted to each language.

For all roles, the agent strategy to perform all kinds of actions (like voting or accusing other players) has its basis on a belief points system. According to the other users' utterances in natural language, Dreaming updates belief points such that the agent with the most points is the most believed (last one to be voted and the first one to be supported) and the one with fewer points is the least believed (first one to be voted and to be accused). The system updates points each time it receives utterances from other players. The belief points update criteria vary depending on the current role of the agent. For example, if Dreaming is a werewolf, it will give more belief points to agents more likely to be the possessed (like possible fake seers). On the other hand, if the agent is, for example, a villager or a seer, it will give fewer points to people likely to be the werewolf or the possessed. When voting takes place, the system selects candidates to vote from the players with fewer belief points and, in case there are more than one, the most voted player in the last night is selected (to take into consideration other players' actions). The seer is the only role that can vote also considering veridic information from its divinations. The seer divines the most suspicious players (the ones with fewer belief points) first. And the werewolf attacks the players less likely to be the possessed. The werewolf and the possessed also have the ability to fake a seer in case no more than 1 seer has come out yet (to avoid having more 3 seers).

- Dreaming is a retrieval-based dialogue system with utterances belonging to different categories:
- Greeting. Ex. Good morning! Did your dreams come true?
- Coming out. Ex. Everyone, wake up! I am coming out as a xRESULT!
- Divination. Ex. While I was dreaming, I divined Agent[0xID] and it seems to be a xRESULT.
- Ask a question. Ex. »Agent[0xID] Who do you think is the most suspicious?
- Unknown response. Ex. »Agent[0xID] I don't know what are you trying to tell me.
- Defense. Ex. »Agent[0xID] That is not true. I am a human!
- Thank. Ex. »Agent[0xID] Thank you for believing in me!
- Accuse. Ex. I think we should vote for Agent[0xID].
- Show trust. Ex. Let's believe in Agent[0xID]!
- Think. Ex. »Agent[0xID] Okay, I will think about that.

- Other. Ex. Well, I will just keep dreaming.

The system customizes utterances during the game to refer to different agents (Agent[0xID]), to present different information (xRESULT, which can be a specific role or "Human"), and to directly talk to another agent (»Agent[0xID]).

The following category priority order is followed by the system when talking is possible:

1. Greeting at the beginning of each day.
2. Coming out in case there is the need to do so (ex. if the agent is the seer).
3. Divination in case there is the need to do so (ex. if the agent is the seer or wants to fake it).
4. Defense in case the agent detects an attack from another user.
5. Thank in case the agent detects a support message from another user.
6. Response to direct messages from other players.
   - Defense in case of an Attack.
   - Thank in case of a support message.
   - Think if the message contains an attack or a support message referring to another player.
   - Attack in case the question asks about who should be voted.
   - Unknown response in case of not understanding the question.
7. The system randomly selects a message from the following categories if possible:
   - Accuse another agent (can be repeated once on the same day).
   - Show trust to another agent (can be repeated once on the same day).
   - Ask a question to another agent (can be repeated multiple times in the same day).
8. Other message is sent if the game has advanced enough.

The system tries to categorize other agents' utterances using keyword searches so that it can provide appropriate responses. According to the target language of the game (English or Japanese), Dreaming uses different utterances and considers different keywords when processing the content of the other players' messages.

2.3 forestsan

Team *forestsan* aims to create their system that can survive until the end of the game by not collecting attentions from other players. For this purpose, their agent pays attention to the other agents to relatively reduce attentions from other agents. This is performed by putting questions to other agents. Dialog analysis is performed by regular expressions.

Their utterance generation algorism is as follows. When there is any question to their agent, they generate a generic response e.g. "I won't tell you", "It is you". When there is no question, their agent generates a question to other agents, or generates role specific utterances e.g. coming out roles.

In the vote turn, they decide their vote target by seer's role coming out utterances. When there is any agent specific behavior, they use such characteristics as well.

When the agent's role is a villager, and if there are three seers come out, then decide their vote target among these three seers. If they could infer the true seer, then vote to the same agent as the true seer.

When the agent's role is a possessed, they decide their vote target from other seers. They always come out as a possessed in Day 2. When they know who is a werewolf, they vote to other agents but not to the werewolf.

When the agent's role is a seer, they always come out their role. If they obtain werewolf by divine result, they always vote to the werewolf. If there is two or more other (fake) seers, then vote to one of these seers.

When the agent's role is a werewolf, they decide their vote target from seers. If there is any possessed survives in Day 2, they come out as a werewolf and tell they know who is the possessed.

2.4 Kanolab

Team *Kanolab* focuses on a genuine seer and a fake seer. They implemented their player agent system that can make inferences depending on the progress of the game, defining role patterns based on the utterances of the genuine and fake seers. Refer to [4] for details.

2.5 Udon

The agent of Team *Udon* aims to play with humans naturally. They focus on three points: their agent behavior could be affected by other agents, their agent could have been felt like having personality, and their agent could tell their reasons.

They convert input natural language into the AIWolf protocol first. When another agent generates utterance that infers some role, following three actions could happen: agree to the inference, suspect

the agent, or believe the agent. These actions express success and failure of persuasions that could allow manipulating other agents' opinions when playing with human players.

They generate utterances from their inspection results, opinions of vote targets and inferences. They generate reason utterances of vote targets and inferences from the highest score reason.

Their agents have five parameters of Egogram for characterizations. For example, an agent of higher tolerance tends to believe villager inferences of others, an agent of higher adaptability tends to adapt to other opinions without spontaneous opinions, etc.

3 Shared Task Runs and Evaluations

All of our shared task runs are in a five players werewolf games as described in Section 1.

Our shared task runs were performed in *self-matches* and *mutual matches*. The same five player agents play games in the *self-matches*; different five player agents play games in the *mutual-matches*. The shared task reviewers are required to perform subjective evaluations based on game logs of these matches. The game logs will be available from the workshop website [3].

We performed subjective evaluations by the following criteria (Table 1):

Subjective evaluation items (5-level evaluation)	
A	Natural utterance expressions
B	Contextually natural conversation
C	Coherent (not contradictory) conversation
D	Coherent game actions (vote, attack, divine) with conversation contents
E	Diverse utterance expressions, including coherent characterization

Table 1 : Evaluation Criteria

This subjective evaluation is based on both self-match games and mutual match games. This subjective evaluation is same as the evaluations in the previous AIWolf natural language contests. Table 2 shows the evaluation results.

4 Conclusion and Future Work

We hold the AIWolfDial 2019 shared task, where five participants provide agent system both in Japanese and English that play the conversation game "Mafia", or the Werewolf game. We performed subjective evaluations based on the game logs of self-matches and mutual-matches. We plan to continue this shared task series in the next year.

Name	Lang	Total	A	B	C	D	E
CanisLupus	JA	3.52	4	3.2	3.4	3.6	3.4
Dreaming-ja	JA	2.72	2.6	2.4	2.6	3.2	2.8
Forestsan	JA	2.68	2.4	2.6	3.2	3.2	2
Kanolab	JA	3.4	3.2	3.4	3.4	3.6	3.4
Udon	JA	4	4	4.2	4	4	3.8
CanisLupus	J-E	3.93	3.33	3.66	3.66	5	4
Dreaming-en	EN	3.20	3.33	2.33	3.66	3.33	3.33
Forestsan	J-E	2.13	1.33	1.66	2.66	2.66	2.33
Kanolab	J-E	2.00	2.33	2	2.66	2.66	2.66
Udon	J-E	3.06	2.66	3	3	3	3.66

Table 2 : Evaluation Results
JA, EN, J-E stand for Japanese, English, machine translation, respectively.

Acknowledgments

We wish to thank shared task reviewers for performing the subjective evaluations, and the members of the Kano Laboratory in Shizuoka University who helped to run the shared tasks. This research was partially supported by Kakenhi.

References

[1] Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Driesshe, van den G., Schrittwieser, J., Antonoglou, I. Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham., J, Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., Hassabis, D., "*Mastering the game of Go with deep neural networksand tree search,Nature"*, 2016, Vol.529, No.7587, pp.484-489

[2] Toriumi, F., Inaba, M., Osawa, H., Katagami, D., Matsubara, H., Kano, Y., Otsuki, T., Sonoda, A., Minowa, S., Aranha, C., *Artificial Intelligence based Werewolf*, http://aiwolf.org/

[3] Kano, Y., Aranha, C., Inaba, M., Toriumi, F., Osawa, H., Katagami, D., Otsuki, T., *AIWolfDial2019*, https://aiwolfdial.kanolab.net/home

[4] Tsunoda, I, Kano, Y. AI Werewolf Agent with Reasoning Using Role Patterns and Heuristics. AIWolfDial 2019 workshop, INLG 201